Information Display Concepts
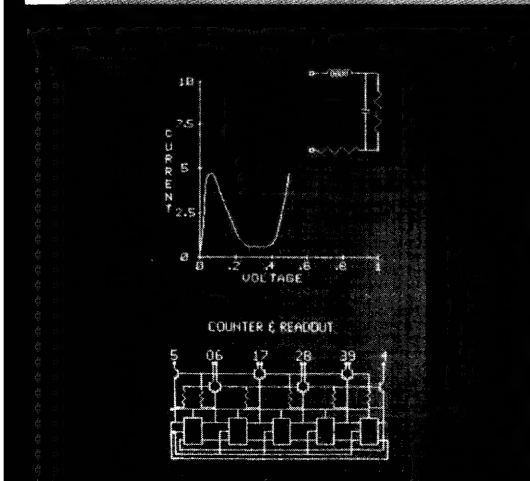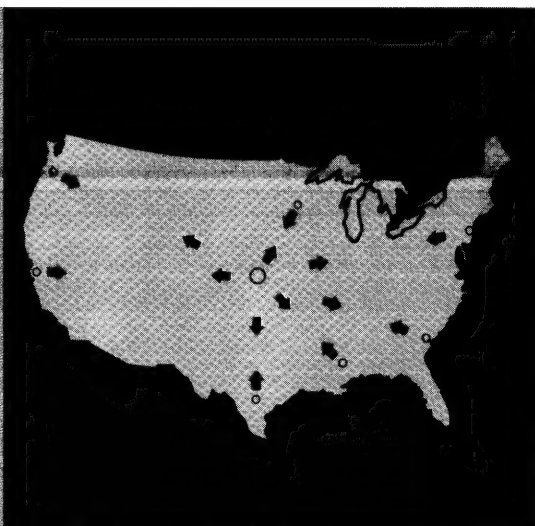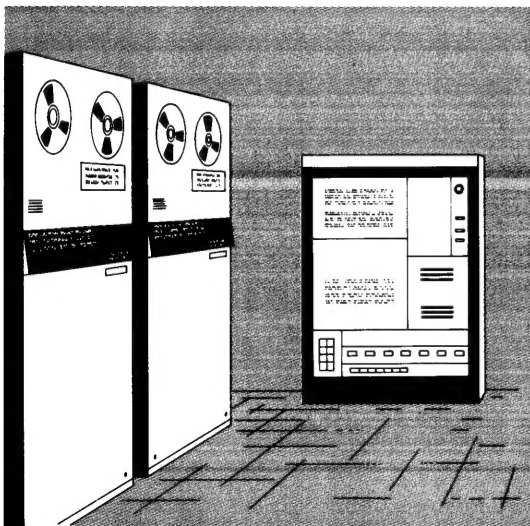
INFORMATION DISPLAY CONCEPTS

BY
NICK STADTFELD

Significant Contributions

by

BOB LeBRUN

TEKTRONIX

CONTENTS

1

INTRODUCTION

An information display unit is defined as "a contrivance for exhibition of intelligence." Just a windy way of saying "a unit for displaying information."

Probably the first information display device was a bare patch of ground upon which a cave man scratched an arrow pointing to water, food or the closest cave. The patch of ground became a wall, and then a chalk board. As writing evolved, civilized man used scrolls, then paper as the medium of display. Mechanical typewriters and printing presses replaced the written hand. Electricity brought more sophisticated mass media in the form of light projection systems, motion pictures and neon signs. Finally, the electronic age has brought radio, television, the oscilloscope and other cathode-ray tube inventions to the field of display devices. With these devices, electronics has brought the computer.

This book will be primarily concerned with those display units and the interconnecting systems used to exhibit information and communicate with the electronic computer.

Chapter 2 describes the operation of the peripheral devices that are used in *the computer room* to provide communication with the computer.

Recently computers are being put to work in a new way, known as "time-sharing." Time-sharing allows the computer to have many simultaneous users; each located away from the computer site. This new application calls for new display techniques.

Chapter 3 is about time-sharing and what is involved in a typical time-sharing system and its displays.

Chapter 4 is an attempt to provide a humble background in the complex subject of computer programming. Programming is so essential to computer operation, that a basic understanding is required for further study of time-sharing and remote display control.

Chapter 5 is a detailed description of digital data transmission as used with time-sharing systems. The intention is to provide a basic understanding of how the data gets from the computer to the remote terminal.

Time-sharing remote terminals are of several different types. Chapter 6 provides background in the three most widely accepted terminals in use.

The terminal must be able to "talk back" to the computer to provide two-way communication. Terminal output devices are the subject of Chapter 7.

Since the display cathode-ray tube usually requires analog deflection voltages, the digital information from the computer must be converted before it is usable. Chapter 8 discusses digital-to-analog, and analog-to-digital conversion. Special digital-to-analog units known as "character generators" and "vector generators" are described.

Chapters 9 and 10 deal specifically with the Tektronix storage display units. Chapter 9 is a treatise on the *storage tube* characteristics and specifications. Chapter 10 deals with the electronics that drive the storage tube.

The depth of coverage of these subjects is intended to provide a working vocabulary and a very basic understanding of the use of cathode-ray tube displays with the computer.

2

LOCAL COMPUTER PERIPHERALS

For the human being to use the computer, he must be able to communicate with it. To communicate, the computer must be equipped to receive inputs and make outputs. The computer can be directly connected to a number of "peripheral" machines whose function it is to accept inputs and make outputs. Today these machines are usually located in the computer center close to the computer's central processing unit (CPU). This chapter will make the assumption that the input/output equipment discussed is in the computer center. Remote computer terminals will be considered in detail in later chapters. The distinction is an important one.

buffer
storage
register

Since the computer is capable of sending data much more rapidly than the mechanical peripheral gear could possibly respond to, a buffer storage register is used to store data, while the computer's control section determines the speed with which the data is to be sent. The speed of the input/output (I/O) operation is usually written into the computer's program, and is acted on by the control section through "interrupts." The buffer storage register is used as the distribution center for all I/O operations. If outputs to two output units, say printer and card punch, are called for; then the computer will output to both at a rate that can be handled by the slower. Depending on the computer, each peripheral device may be handled independently.

program
interrupts

The computer handles input/output operations through "interrupt" priorities spelled out in the computer's program. An "interrupt" is a signal from a peripheral unit that the unit is ready to send or that it is prepared to receive. Interrupts are coded in order of importance (priority) by the computer program.

The interrupt system allows simultaneous operation of the I/O devices with central-processing-unit operations. When the computer I/O buffer receives an interrupt, it notifies the central processing unit. The program interprets the priority of the interrupt and acts accordingly. If it is a high priority, for instance, the computer will stop what it is doing, initiate a routine to respond to the interrupt (called "branching"), process the interrupt data and then return to where it was before interrupt. If it is a low-priority interrupt, the computer might continue what it is doing until it is convenient to exit to service the peripheral unit's request. This interrupt system is important for efficient use of the computer's and peripheral equipment's time. The computer doesn't want to be jumping around randomly, nor does the external equipment want to wait unnecessarily long.

The need for priorities is immediately evident when you consider that the computer could receive interrupts from two or more peripheral units simultaneously. The program must decide in what order they should be processed. Not only does interrupt priority determine order of execution, but it also decides the program branch, and the unique subroutine to be used. Sometimes more than one peripheral unit has the same priority level, and therefore uses the same branch subroutine. In this case it is necessary, by programming means, to identify the individual requests causing that interrupt priority level to be energized.

While a high-level request is being executed, the machine will not recognize lower-level interrupts. However, if a request is detected for a higher priority level than is presently in process, the program is immediately interrupted again. This is frequently called "nesting" of interrupts.

input/
output
devices

The most common local computer input devices are the console lever switches, card reader, typewriter, magnetic-tape reader and punched-paper-tape reader. Output units include paper-tape punchers, card punchers, magnetic tape, a variety of printers, typewriter, mechanical X-Y plotter and cathode-ray-tube display.

A brief discussion of the terminology involved and operation of each of these units follows. An attempt will be made to include how the information is transferred to and from computer and peripheral machine.

punched-card
reader

Perhaps the most widely used device to input data to the computer is the punched card. The use of the punched card actually requires two machines. The "keypunch" machine looks like a large typewriter and is not connected to the computer. When the keypunch operator presses a symbol "key," one column of the card is punched in a pattern representing that symbol (letter, number or special function). Each vertical column of the card has 12 areas where holes may be punched. A "hole" or "not hole" then represents a "one" or "zero" in a 12-bit "word," that is the code for that symbol. Most cards have 80 columns, so that 80 symbols or "computer words" can be punched into a single card.

Once the data has been converted into holes in the card, the cards are entered into the "card reader." The reader may sense the holes either by row, column or entire card. The sensing is done by passing the card under brush contacts or photocell light pickups. If column sensing is done, each 12-bit column is converted into a 12-bit word that is entered into a data register in the reader. The card reader's data register is electrically connected, by 12 parallel wires, to the computer's central processing unit. The card-reader's data register then transfers, in parallel, to computer memory, the 12-bit word containing the code for the symbol punched on that card column. The column-by-column read operation and register transfer is controlled by interrupts, and commands from, the central processing unit.

The terms "bit," "computer word," "register" and "parallel transfer" are discussed in detail in Chapter 5.

The keypunch operator punches the cards at mechanical speeds at the keypunch machine, but once the cards are prepared, the stack of cards (called a "deck") can be read into the computer at a rate of 100 to 1000 cards per minute. Five-hundred cards per minute is equivalent to 40,000 symbols per minute, or 480,000 bits per minute.

Computer data can be read back out to the card reader
which has a punching unit able to punch cards on
command from the computer. Thus, a "stack" can be
made by the computer. This is usually done to
preserve the program on cards for future use, or in
case the program is inadvertently changed or
destroyed in memory. The punching operation can
take place at about 250 cards per minute.

magnetic
tape

The read-in speed that the card reader is capable of,
is not fast enough for many applications. Magnetic
tape is used as an intermediate input medium. The
information is usually entered into the computer from
cards, and then transferred to magnetic tape.
Subsequent operations call the information directly
from tape without going back to the slower cards.

Today, the magnetic-tape unit is becoming an input/
output device with usage similar to the card reader.
Machines are available to enter data directly on
tape from a keyboard. These machines are relatively
new and their use is not yet widespread. Magnetic
tape finds acceptance as memory because of its low
cost-per-bit of storage capability.

Magnetic tape is coated with an 80-percent iron-
oxide material in which small areas can be magnetized.
The tape is in motion and can be thought to be made
up of imaginary channels. Each channel is the width
of a recording head. The digital information in
the channel is bounded by imaginary lines forming a
magnetic field zone called a "cell." The fundamental
unit in a binary numbering system is the "bit" and
each cell contains one bit of data. Tape widths vary
from 1/4 to 1 inch. The most popular size is 1/2
inch. The 1/2-inch tape typically has seven
magnetizable cells across its width. A seven-bit
word can thus be stored at one position, and more
than 1000 of these words can be stored on one inch
of tape. Millions of words can be stored on a reel
that holds between 2400 and 3600 feet of tape.

The magnetic-tape unit contains "read" and "write"
heads. "Read" means to output to the computer.
"Write" is to input onto tape. The write head is
divided into a number of elements, each of which can
be activated to create a magnetic field. In a write
operation, the computer sends pulses or levels to
the write head to activate the elements to a "field"

or "reverse field" state. As the tape passes the
head, the spots on the tape are magnetized or reverse
magnetized, depending on the condition of the fields.
Thus the computer word is stored on the tape. In a
read operation, the tape is pulled past the read head.
The head is sensitive to field changes created by the
magnetic state of the tape's cells. As the cells
are pulled past the read heads, the heads generate
plus and minus pulses or DC levels, depending on the
state of the magnetic cells on tape. The bits are
sent to computer memory as with the card reader.
Magnetic tape can read out 80,000 characters per
second, about 120 times faster than a card reader.

The direction of magnetization of the tape "cell"
determines whether the magnetic area contains a
"one" or "zero."

RZ and NRZ
recording

There are a number of methods of coding digital
magnetic tape, each with advantages and disadvantages
too involved for this discussion. The coding
methods you are most likely to hear about are pulse-
signal recording, often called return-to-zero (RZ);
and DC level recording, called nonreturn-to-zero (NRZ).

The RZ-type recording requires a pulse to turn on
(zero to saturation) and a pulse to turn off
(saturation to zero). Since the tape can be
saturated both positively and negatively from zero,
pulse-type recording may have three signal levels.
A pulse is used on each cell of the tape.

The NRZ recording signal type is a DC-level change.
The levels available are plus saturation and minus
saturation. There is no zero level. Hence the
name "nonreturn-to-zero." When a recording head is
signaled to the plus saturation level all tape areas
passed under that head will saturate in the
corresponding direction. A level change is required
to get a bit to record in the opposite direction.
If alternate ones and zeros are to be written, level
changes are required between cells, and the waveform
would then resemble RZ recording.

There are a number of variations to the NRZ-coding
method. IBM has used a method called "NRZ MARK" or
"NRZ Module 2" so extensively that it is often
referred to as the IBM-coding method. The technique
is too involved to be discussed here.

magnetic discs — The magnetic disc and magnetic drum are used as intermediate storage, in the same manner as magnetic tape. The magnetic disc looks very much like a phonograph record. It rotates at high speed. A number of these discs, 10 to 20, are placed on a spindle. Information is then read on these discs simultaneously. The disc and drum have a faster retrieval time (access time) than tape because of the small size and rapid rotation of these units. Any point on them can be rapidly reached, whereas the magnetic tape has to be wound and rewound.

magnetic drum — The magnetic drum makes use of a magnetic surface, similar to that of tape, on a cylindrical drum. As the drum rotates, each area of the surface comes under one of a row of read-write devices along the drum. At the time that a given area is under the read-write heads, bits can be read off or recorded into the surface. The bits are thus pulled off with parallel heads reading many tracks as "words" as the drum rotates at high speed.

The digital bits are usually stored on the tape, disc or drum in such a way that a computer word cannot be read out in one parallel operation. For example, words are usually stored on tape in a serial-parallel fashion. If the computer word is 28-bit-positions long, 28 cells would be required. Since seven channels are available across a point on the tape, the word would be stored on four tape positions of seven bits each. These units require

format translator — a "format translator" to assemble data bits into computer words in a read operation, and to break the computer word up in a write operation in such a way that it can later be reassembled.

console switches — Most computer consoles have indicator lights for the "states" of the internal registers, and lever switches or buttons that can be used to control those states. Information can be entered word-by-word, by hand setting these switches and activating a "store" button. This is the slowest and most tedious method of input, and is used only for initial addressing or troubleshooting.

keyboard and printer — Information can be input to the computer through a typewriter-like keyboard. The unit is usually supplied as part of the computer console. The keyboard has 53 character and function keys; most are

standard typewriter symbols. The keys, when pressed, generate a binary-coded decimal (BCD) or straight binary code that is transmitted in parallel form to the central processing unit. The setup is such that the computer then sends back the code to activate the printer, to type on paper one character at a time like a typewriter. The printer, used as an output from the computer, prints 6 to 20 characters per second. The keyboard and printer are hooked up in one unit closely resembling a typewriter. Because the computer operates in *straight binary*, the computer programming must code for transmission, and decode for reception of the *binary-coded decimal* signals.

The keyboard and printer used in the console are slow and limited in usefulness to small amounts of data. To print out at high speeds; line-at-a-time, on-the-fly and matrix printers are used.

Line-at-a-time printers print a row of characters, rather than a single character. There may be 25 to 120 character positions (columns) per row. A buffer store in the unit holds the complete line of symbols to be printed. The speed of these units varies from 100 to 150 lines per minute.

The line-at-a-time printer is limited by the time required to accelerate the type bars or wheels. The on-the-fly printer overcomes this limitation by keeping the internal mechanism rotating. This technique develops printing speeds from 300 to 900 lines per minute.

high-speed printers
The very high-speed matrix printers form characters not by type font (a complete assortment of printing type in one size and style), but according to a dot pattern selected from a rectangular dot array (5 X 7). The actual printing takes place by small wires that transfer ink to the paper by striking it. Matrix printers are capable of 500 to 1000 lines per minute.

Transmission of data from computer to printer is usually done in parallel form. The buffer storage register is the distribution center. Control of the operation is from the central processing unit, under direction of the program, through the use of interrupts.

$$Y = 0.105E\ 02 + -0.535E-01 \cdot X**1 + 0.370E-01 \cdot X**2 + -0.113E-02 \cdot X**3 + 0.104E-04 \cdot X**4$$

POLYNOMIAL REGRESSION EQUATION
X =YEAR(TWENTIETH CENTURY)
S = 1.30
R = 0.99

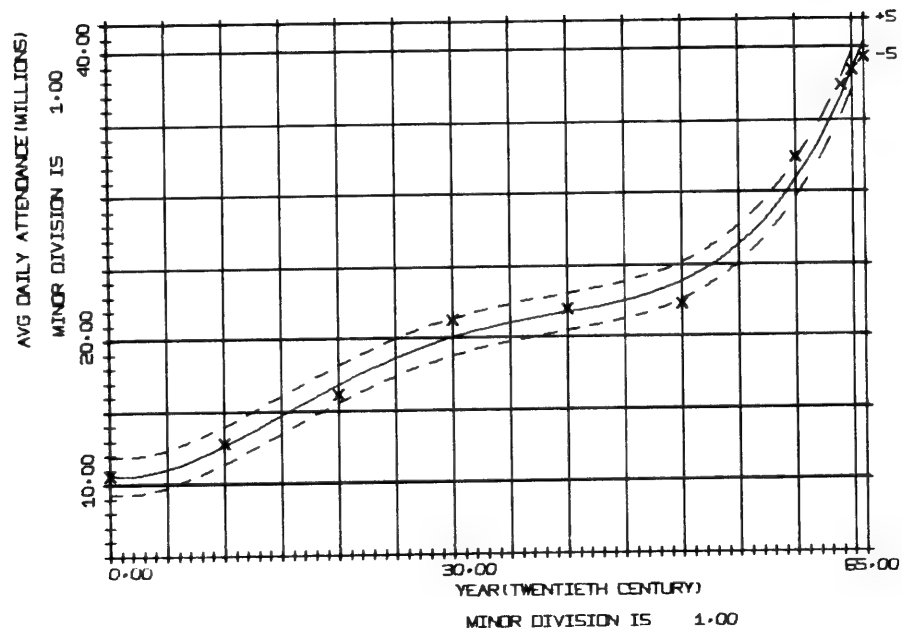AVERAGE DAILY SCHOOL
ATTENDANCE VERSUS
YEAR

Fig. 2-1. Plot output from California Computer Products Inc. 1627 plotter.

graphical
output

Many times the output information provided by the computer can best be presented in graphical rather than alphameric form. For example, the computer can accept data on the logic required to perform a given circuit function. Knowing the logic, the computer, when properly programmed, can design the circuit--make component selection, interconnection decisions, etc. The engineer can use this data to draw the schematic diagram. The schematic is the desired result, the intermediate calculations and subsequent answers are one step from that result. With the help of a plotter, the computer can make that last step.

The computer can be programmed to make outputs to peripheral equipment that will plot points, thereby drawing the schematic directly.

The three units used for graphical plotting are the multi-stylus recorder, the X-Y mechanical plotter and the cathode-ray tube.

multi-stylus
recorder

The multi-stylus recorder consists of a data register, numerous styli (as many as 1024) and a moving roll of recording paper. The styli do not move back and forth as in a normal recorder, but instead are fixed in a row. Their only motion is to write or not-write on the paper. (Stylus up, stylus down.) As the paper is pulled past the row of styli, they write or not-write depending on the digital information in the data register. The resolution of the recorder depends on the number of styli used. The multi-stylus recorder is much faster than the mechanical plotter since a number of points can be plotted simultaneously. The electronics is simple, because the styli are driven directly from the data register.

Since the paper only moves in one direction, the plots have to be sequenced in time. This greatly complicates the plot programming. The multi-stylus recorder is more expensive than the mechanical plotter and not nearly as widely used.

mechanical
plotter

Mechanical plotters are available to use the digital data directly from the computer to draw coordinates, plot data points, connect them to show the maximums and minimums, and plot trend lines. See Fig. 2-1, for example. The California Computer Products, Inc. 1627 plotter is such a unit.

Often the plotting information is stored by the
computer on magnetic tape. The tape is then played
back "off-line," at a data rate compatible with
the mechanical plotter. This involves very low tape
speeds ($\cong 3$ inches per second) to supply data at
300 increments per second. At this rate an 8-1/2 x
11-inch plot may take from 3 to 15 minutes, depending
on its complexity.

Mechanical plotters come in a variety of sizes; from
units that will draw displays on an 8 1/2 x 11-inch
page, to those large enough to make a full-scale
drawing of the wing of a giant airplane. The smaller
plotters are more economical and more widely used.

plotter
operation
The plotter executes under digital command from the
program, controlled by the central processor. See
Fig. 2-2. The actual recording point is set up by
the independent or combined incremental motion of the
pen or the paper beneath it.

The pen is mounted on a carriage that is capable of
horizontal motion. The paper is moved vertically,
by rotating the pin-feed drum in a forward or
reverse direction. The drum also functions as a
platen. The pen can be raised or lowered, to write
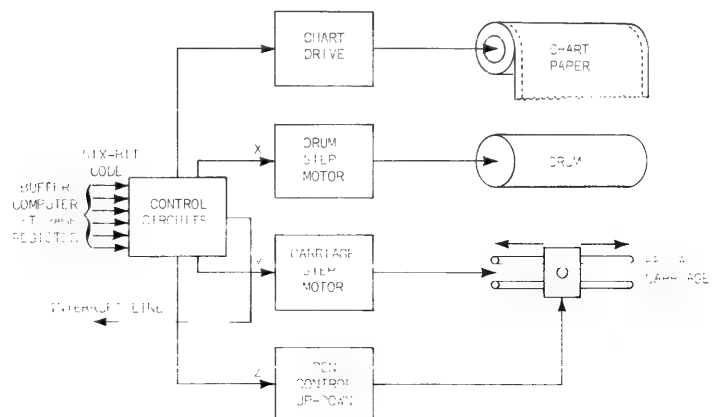or not-write on the paper.

Fig. 2-2. Mechanical plotter.

Each command to the plotter causes the carriage and/
or drum to increment 1/100-inch (some plotters use
1/200 or 1/500-inch) or the pen to raise or lower.
Combined commands to carriage and drum produce a
resultant diagonal. Carriage and drum movements
require about 2 to 5 ms of plot time. Pen motion
takes 100 ms.

The plotter employs a bidirectional rotary step motor
on both X and Y axis drives. The motor is capable
of 300 steps per second. The pen motion is controlled
by solenoid.

Six parallel data signals from the computer control
all action of the plotter. These pulses arrive as
a six-bit "word" on parallel lines. Each line has
control of a part of the plotter operation. One
line causes drum-up, another drum-down; two other
lines initiate carriage-right and carriage-left, while
two others are for pen-up and pen-down. The motion
of the plot related to the control digits is seen
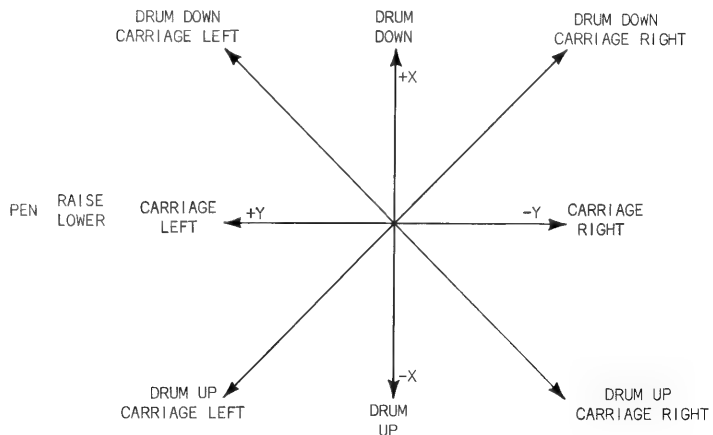in Fig. 2-3.

Fig. 2-3. Direction of movement related to control
           code as viewed from front of plotter.
           Graphs are plotted with axes oriented
           as shown.

The computer knows when to send to the plotter through interrupt signals. When ready to plot, the plotter sends an interrupt signal to the computer I/O buffer. The interrupt signal is a specific state of a flip-flop. The computer program recognizes the interrupt and sends the six-bit position information to the plotter. The information also flips the interrupt flip-flop, which removes the interrupt signal. The flip-flop remains in this state long enough for the plotter to complete its move. It then flips back, sending another interrupt to the computer, saying "I'm ready."

The plotter can only move from point to point in 1/100-inch steps, whether it is writing or not. A computer word is required for each step. There is no provision for a long step to skip over blank areas in the plot with a single computer word.

Great care is taken to maintain accurate registration of the carriage and drum. Thus drawings and plans can be accurately drawn to scale.

CRT
displays
Either a refreshed or bistable storage cathode-ray-tube display (see Chapter 6) can be used as a printer or plotter of output data.

The CRT is faster than most printers and much faster than the mechanical plotter. Typical tape transports have maximum data rates from 15 to 60 kHz. Computers themselves have data rates out of core memory from about 100 to 500 kHz. Since the mechanical plotter can accept a rate of only 300 Hz, the magnetic tape and computer range from 50 to 2000 times faster than the mechanical plotter. With a settling time plus dot-writing time of 25 μs, the storage display unit is capable of data rates of 40 kHz, about 130 times faster than the mechanical plotter.

There are four limitations to the average storage cathode-ray tube when used as a plotter. The resolution of the CRT is less than the plotter, the linearity (line straightness) is less, the plot size is limited by CRT size, and hard (paper) copy is more expensive.

The Tektronix Type 611 Storage Display Unit, when compared to California Computer Product Inc.'s 1627 plotter lends some magnitude to these differences:

	1627 PLOTTER	611 DISPLAY UNIT
Information display rate	300/dots/second	40,000 dots/second
Resolution	100 line-pairs/inch	50 line-pairs/inch
Linearity	.1%	2%
Display range	30" x 120'	16 x 22 cm
Hard copy	8-1/2 x 11" plot has $.01 paper cost	4 x 5" Polaroid is $.30 per picture

For these reasons the cathode-ray-tube display will not *replace* the mechanical plotter for most applications in the computer center, but, rather would complement it nicely. The CRT display, being so much faster, performs superbly as a quick-look or fast-check device. In this way, a check can be made on the accuracy of the plot or plotting program and changes made, or plot scrapped, before *minutes* of computer time are wasted while plotting.

The CRT display is also useful for checking the punched tapes used for operation of numerical control machines. An error in a numerical control tape could cause a machine tool to incorrectly punch a hole or mill a part. If the tape is run into the computer, and the numerical control machine's path of travel plotted on the CRT, the tape can be verified.

In the mechanical plotter, it was necessary to convert the six-bit computer code to voltages to step a motor. The motor, in turn, mechanically positioned the drum or carriage. In this way, the computer digital data was converted to a spatial plot.

The cathode-ray tube requires X and Y analog voltages
to deflect the beam, and Z-axis voltage to turn it
on or off. Four of the six bits from the computer
must be converted to incrementing analog voltages to
position the CRT beam, and the two bits used for
pen control in the mechanical plotter are used to
turn the beam off and on. This is done in a plotting
interface unit, located with the display unit. The
CRT uses the same plotting program routines as the
mechanical plotter, except that software changes are
made to speed up the operation. Basically, this
consists of eliminating the computer requirement of
waiting for an interrupt from the plotter, before
sending another positioning code. With some computers,
the CRT display is fast enough to plot points as
fast as the computer I/O buffer register can send.

Although the CRT display unit uses almost the same
software as the plotter, the output is usually via
a separate channel, rather than the plotter channel.
This allows high-speed CRT plotting, at the same
time that the plotter is operating. The output
channels are given names by the manufacturer, System
Access Channel (SAC), Multiplex Channel, Selector
Channel, etc.

The CRT-display-plotting interface can be equipped
with circuitry to expand portions of the plot to
show greater detail. The areas to be expanded
are selected by the operator, and the plot is rerun.
Operation is somewhat like sweep expansion on an
oscilloscope, in that portions of the total plot are
off-screen, but are still being plotted by the
computer.

3

TIME SHARING

In Chapter 2, we discussed the use of display devices
in the computer center. These displays have played
a large part in computer development, for a computer's
success depends on its ability to accept and answer
questions. With the help of these display devices,
the computer has, as promised, become a brilliant
tool. It can plot a space rendezvous, design its
own logic circuits, draw up payrolls and control a
refinery operation -- all in a fraction of a second.
But the man who asks the questions may wait hours
or days for answers, calculated in microseconds,
to sift down through layers of experts and office
routine. The Data Processing Department surrounding
most computers limit the computer's usefulness for
applications requiring immediate answers or running
dialogue between computer and man. Real-time usage
has been possible only by devoting one high-speed
computer to one slow-thinking man, a wasteful and
prohibitively costly use of the computer for
practical purposes.

Today, "real-time intimacy" is evolving through the
use of "time sharing," one of the most important
developments in the use of computers. By sharing
the users and the costs, the time-sharing computer
distributes computer services at truly computerized
speeds. Like a doctor making his hospital rounds,
the computer switches from one user to the next, at
intervals measured in milliseconds. By sharing the
costs along with time, the expensive computer is
put within the reach of a modest budget.

The significance of time sharing is illustrated best
by the changes in how the computer is *used*, rather
than the changes to the computer itself. By the
batch
processing
"batch processing" method, the computer handles one
problem at a time. The problem is taken to the
programmer, who gathers problems from many users,
compiles a long program and feeds information via
punched cards into the computer, which spits out
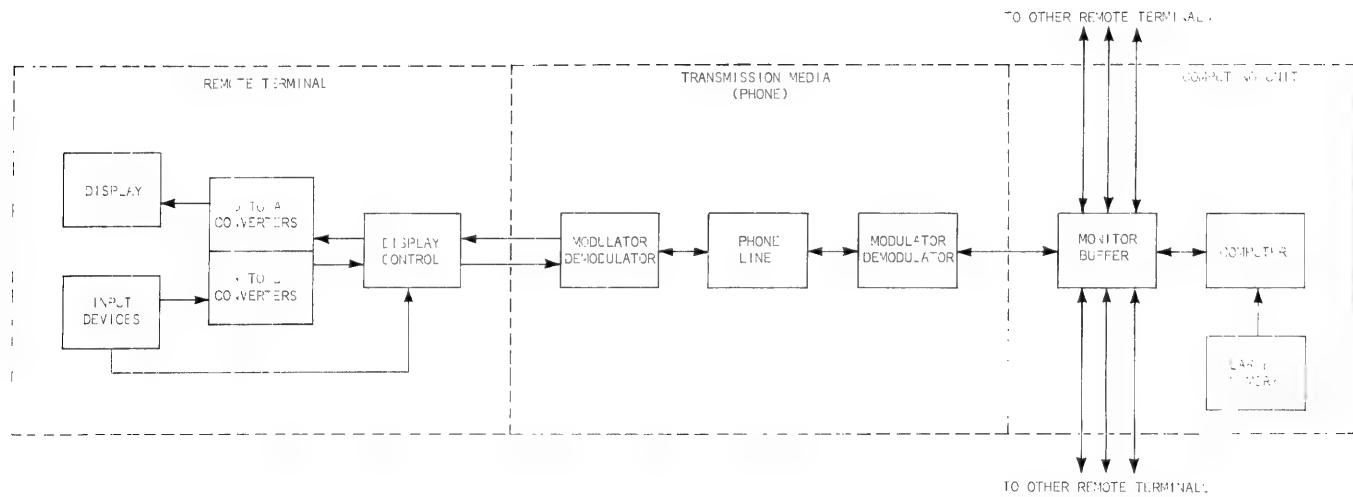the answers in sequence.

Fig. 3-1. Time-sharing system block diagram.

But the user does not see the answer yet. It comes
to him hours, maybe days, later via conventional
paper distribution. If the problem was stated
incorrectly or an error used in the program, then
the problem must be corrected and run through the
entire sequence. Complex programs may require many
iterations, since the first error often prevents
debugging of later parts of the program.

time
sharing

In contrast to batch processing's tedious procedures,
time sharing permits the user to deal directly with
the computer. He sits down at his console and
keyboard in his home, office or laboratory; dials
the computer as he would dial a telephone number, and
then types out his questions. There are no people
or procedures between him and computer; he
questions the computer directly and answers are
returned directly.

This "real-time interaction" together with high-
order languages, frees the user from having to ask
formal questions and, in due time, get formal
answers, as in the batch method. Instead, he can
ask, in direct dialogue, unplanned questions that
occur to him during the dialogue and that were not
foreseen in the original program. It also means he
can debug a new program quickly, because bugs become
immediately evident. The computer, in effect,
becomes the user's intellectual assistant.

At first, time sharing was thought to be only a
convenience; a means of allowing fuller use of the
computer by more people and of saving time and cost
for the users. In practice, experiments with the
technique have demonstrated that the "continuous
dialogue" effect holds the greatest possibilities.
The system makes it possible for users to carry on
a discourse with one another through the machine,
drawing on a large store of knowledge as they do so.
The system can unite a group of investigators in a
cooperative effort, or it can serve as a pool of
knowledge and skill on which anyone can draw according
to his needs. On a large scale, one can conceive of
the time-sharing system as an extra-ordinary,
powerful library serving an entire community -- in
short, an intellectual public utility. In fact,
time-sharing companies are becoming known as "computer
utilities."

A time-sharing system block is shown in Fig. 3-1.
The basic units are the computer, input/output monitor
buffer system, transmission media (usually telephone),
display control, the display digital-to-analog and
analog-to-digital conversion units, the display and
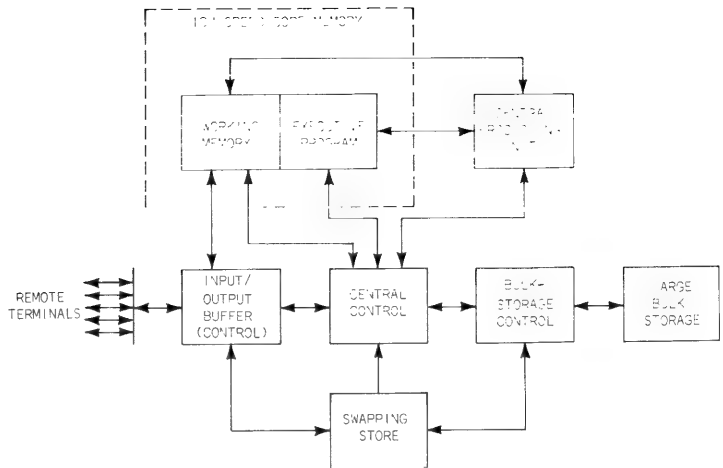the terminal input devices.

Fig. 3-2. Time-sharing computer block diagram.

Lately, computers are being designed especially for time sharing, but modifications to an ordinary computer will do. The physical alterations are technically simple. The time-sharing computer needs a more sophisticated monitor buffer system and large memories to store the programs of its many users. See Fig. 3-2.

When not in use, the user's programs normally reside in bulk storage (magnetic tape, disc files, etc.). When called for, the programs are automatically transferred to a magnetic drum, called "swapping store." They are not sent directly to the high-speed "working memory" because the cost of this memory is presently too dear to permit storage of all active programs. The programs in the more economical swapping store are transferred to and from working memory as required by the "scheduling routine," which is part of the executive program.

The monitor can be likened to the airport control tower, switching and holding data both in and out as the controller does his airplanes.

The "software" required for time sharing is far more important than the hardware items. The computer uses an "executive" or "control" program to supervise the work demands of its many subscribers, who have different kinds of problems to solve in any number of computer languages. The executive program keeps

the whole system coordinated in a sequence determined by priorities. It allots computing time to its many users, brings their programs out of bulk storage and puts them back when the client is through. It also safeguards a user's information, shielding it from other clients, and prevents one client from wiping out another's program. This feature is called "memory protection." At the same time, the executive program keeps track of users, makes corrections, gives help to unskilled users and performs hundreds of other housekeeping duties. All this at speeds which make the individual user think he has exclusive use of the computer.

monitor buffer

The computer monitor buffer performs input/output buffering. The name "monitor" is derived from its function of "monitoring" interrupts from remote terminals. The monitor buffer must convert the serial digital data received from the remote terminal to parallel form for input to the computer and vice versa for outgoing data. It also must store outgoing data should a terminal be busy and unable to receive, and incoming data if the computer is busy. It operates through the program and services interrupts much the same as the input/output buffer storage register discussed in Chapter 2. This monitor system is often another small computer -- necessary to handle the large number of simultaneous messages.

transmission

The information may be transmitted to the distant terminal by many means. The most convenient and readily available are the telephone lines. Microwave links have some acceptance for data transmission and may one day be popular. Artificial satellite transmission or laser networks, both capable of high data rates, may be the medium of the future.

The telephone hookup consists of a modulator-demodulator (MODEM) which encodes the serial digital data on a voice frequency for transmission and decodes at the receiving end.

display control

When the serial digital data arrives at the terminal it must activate the proper piece of terminal equipment and synchronize with what is happening in the terminal. The Display Control performs this function. It accepts data from the computer, terminal keyboard and other terminal outputting devices, and routes the information to the display, character and vector-generation equipment or computer at the proper time.

terminal
D-to-A
and
A-to-D
conversion

The terminal contains the hardware to convert digital
data from the phone demodulator to analog voltages
for production of the display. The digital-to-
analog conversion is often done within terminal
character and vector-generating equipment. These
specialized digital-to-analog converters reduce the
digital data required from the computer by producing
symbols whose dot patterns are "known" to the hardware.

A keyboard and other devices send output data from
the terminal to the computer. Often their outputs
are analog voltages which must be converted to
digital for transmission. The more common terminal-
outputting apparatus are the light pen, X-Y tablet
and Stamford Research Institute's X-Y "mouse."

terminal
display

The terminal display could be a teletypewriter,
refreshed CRT, storage tube or other type of lighted
panel. The display is the user-system interface.
Its ability to effectively present computer data
both graphically and/or alphamerically determines
the effectiveness of the system. The CRT display
accepts X, Y and Z information from the digital-to-
analog conversion equipment and draws lines or dots
to present the data.

Time sharing is helping solve one of the tougher
problems facing the computer -- inaccessibility.
The time-sharing system increases the number of users
and enhances the man-machine interface. Chapters 5
through 10 will deal with the main sections of the
time-share system.

4

PROGRAMMING

To better understand computer operation and, in particular, time-sharing computer operation, a discussion of programming is in order.

A "program" is defined as a sequence of instructions to be carried out by the computer. In conventional computer operations -- not time shared -- the programmer is the person who analyzes the particular job to be done and specifies the sequence of computer instructions to do the job.

How does he do this?

First, he must specify in exact detail the job to be done. Often this is the most difficult part, since it is not unusual to find, upon close examination of a problem, that we do not know what that problem is.

If the problem can be carried out on the computer, the programmer must determine what data must be input to the computer, what the relations between input and output are under all possible conditions, and what the output is to be.

algorithm
The programmer then writes a sequence of specific steps by which the output information can be obtained from the input data. This sequence is called an "algorithm," and is usually written in the form of a diagram, called a "flow chart." The algorithm is not the "program"; but it is a very important step in programming since the choice of sequence will determine how difficult the program will be to write and how much computer time will be required to solve the problem. For economy, both programming time and computer running time should be as short as possible.

Each of the steps in the algorithm is then broken into the specific sequenced instructions that the computer must follow to complete that step. This list of instructions is the "program."

Let's take a closer look at how the programs corresponding to the algorithm are produced.

In the early days of computers, the instructions were written in "machine language", i.e. the program instructions were written in the binary numbers which actually represented the instructions in the machine. This technique is almost never used now because of its slowness and susceptibility to programming errors.

mnemonic
programs

The first step in simplifying programming was to substitute mnemonics for binary numbers representing instructions, and to use decimal integers for the address portion of the instructions. Thus, the programmer writes GOTO 8 instead of 110111001000.

Another simplification was the use of symbolic address instead of the actual computer-memory address location. Before the computer can understand the mnemonic instruction, it must be translated into machine language, and the symbolic address must be translated into machine address. A program written with symbolic address and mnemonics for operations is called "symbolic" or "assembly-language." To translate the assembly-language program into a sequence of machine-language instructions, a program was written called an "assembler" or "translator" which directs the computer while it is converting the mnemonics to machine language.

With a symbolic program, then, the computer must do two things: Convert the program into machine language; then perform the instructions in machine language. The symbolic program is considered a "lcw-level language."

compilers

problem-
oriented
language

An assembler-language program uses one mnemonic instruction for each machine-language instruction. This limitation was overcome through the development of "compilers" and "problem-oriented language." Programmers learned that, in certain types of problems certain sequences of steps occurred frequently. For instance, in mathematical problems, addition occurs

frequently. Consequently, part of compiler-program language could convert the statement "ADD B to C" to a sequence of steps in machine language.

By using an appropriate problem-oriented language, the number of program statements is reduced to less than 1/10 of the machine instructions required. FORTRAN (FORmula TRANslator) is probably the best-known compiler language. It is an example of a "high-level language."

source
program

The input program in a problem-oriented language is known as a "source" program. The process of problem to solution through the various program transformations is shown in Fig. 4-1.

The problem is brought to the programmer, who writes the "source program" in problem-oriented language. The computer translates the source program to a symbolic program, using the compiler program from its memory. The computer then converts the symbolic program to the machine-language program, using the assembler program from its memory. The computer sometimes translates the source program directly into machine-language instructions, i.e. the assembler is part of the compiler. Using the machine-language
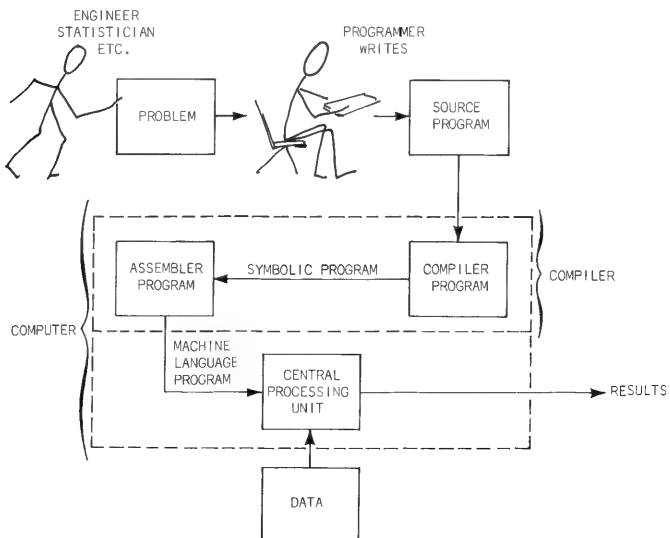

Fig. 4-1. Block diagram showing transformation of problem-oriented language in computer.

program, the central processing unit of the computer then enters the data, performs the instructions and outputs the solution.

subroutines   Programmers often make use of "subroutines." A subroutine is a program that can be used many times within other programs. The programmer stores the subroutine in memory and calls up that memory address whenever he needs to include the routine in his program. The subroutine serves to simplify the programming job and reduces the amount of computer memory necessary to store the program.

program
library

Programs are collected and made available for anyone to use. The existence of this program "library" greatly reduces programming effort because most programs contain parts or are composites of other programs.

software

Most computer manufacturers provide a large number of programs to the customer when they deliver the computer itself. The computer is termed "hardware" and by analogy the program library is termed "software." Since the hardware is of no use without the software, and since the software is vitally important in determining how easily other programs can be written, the quality of software is just as important as the quality of hardware in determining computer usefulness.

debugging

No discussion of programming is complete without mentioning "debugging." It is very difficult for a human to write a program without mistakes. Invariably, the first time a program is run it does not perform correctly. The programmer must then locate and correct the mistakes. Diagnostic programs have been devised which can print out messages to the programmer, pointing out program errors. The computer helps debug itself! Even with this computer help, "debugging" often takes more time than writing the program.

Up until now the discussion has been on how programs are developed. Two other questions need to be answered: *Why* are they written, and *how* are they used?

Programs are written to give instructions to the computer to solve a problem. But why solve the problem by computer? Why not spend your time solving

the problem yourself instead of spending your time
writing a program so the computer can solve it?
The computer's speed enables it to make extensive
calculations in split-second time, but the program
might take hours to set up. To make it economical
to write the program and solve by computer, the
problem is usually one of two types: Those involving
voluminous numerical calculations, that would take
longer and be more susceptible to human error, and
those that may be simple but that occur frequently.
A company payroll is a problem that falls into both
categories: It requires sizable repetitive
calculations and occurs each pay period. No wonder
it was one of the first business problems to be
computerized.

The time and effort it takes to program a problem and
store it in the computer is as much a part of
determining the economics of solving that problem
by computer, as is the cost of the computer time
itself.

Once it is decided that a problem is best done by
computer, and the program is written, it is ready to
be used. First it is entered and stored in computer
memory or in auxillary memory (magnetic tape, discs
or drums). Whenever the user wishes to solve that
problem, he calls his program into memory, if not
already there; feeds in the required input data;
starts the program, and waits for the answers. The
input data consists of the values for all the
variables that the computer must have to solve the
problem.

conversational
language

To free the user from the delays inherent when working
with programmers, keypunch operators and machine
priorities, "conversational" computer languages
were developed. *These are the languages most often
used with remote time-sharing terminals*, although
any computer "language" could be used, as long as
the computer had a compiler and assembler to
recognize it.

"Conversational language" is a problem-oriented
language simplified for users without programming
training. The languages contain compiler and
assembler-program logic and format-error-sensing
loops to detect errors in each input statement from
the user at the terminal. The simplified language

used in the source program generated by the user at
the terminal demands that the compiler and assembler
programs be more sophisticated and occupy more
computer memory.

By understanding a few simple commands and the correct
format, the user at the terminal creates "instant
program." The user writes, runs and debugs his
program as he goes along. The same conversational
language is used to call up and run programs the
user already has stored in the computer. The ability
to "talk" to the computer as each step of the program
is set down permits immediate format and logic
correction of all or any of the program.

The conversational language enables the user to
"consult" with his "assistant," the computer. The
computer is still being told what to do and when to
do it, but as it receives each instruction it can
"talk back" to the user, pointing out format or logic
errors -- telling him it does not understand --
thereby *teaching* him how to talk to it!

The programming language of the future will be vocal.
The user will speak commands into a microphone and
the computer will follow orders. "Solve the equation
$A = X^2 + 2XY + Y^3$, if $X = 1$, $Y = 3$" or "Calculate the
payroll for pay period XXX for ABC Company." Voice-
communication systems are being designed today;
computers can *already* recognize the spoken word.
The man-machine interface gap is narrowing!

5

DIGITAL DATA TRANSMISSION

In Chapter 3, we discussed the total time-sharing computer system. Chapter 5 will deal with its transmission portion. The two main parts of the data-transmission system are the input/output buffer and the telephone transmission network.

In previous chapters we have referred to "transmission" of "digital data." Perhaps some clarification of terms is necessary. Digital data is the electrical representation of symbols (letters, numbers or special functions) or combinations of symbols in the binary state of "ones" or "zeros." The bits (one and zero states) are usually grouped into "words." The original symbol (a device in a particular condition) is something that could be transferred physically. (A page full of symbols could be sent through the mail or a punched card representing a symbol could be moved from punch to reader.) However, the ability to transmit a symbol from one point to the next in microseconds is essential to high processing and communication speed; consequently, there is no time for much mechanical motion. To get this speed, the symbols are converted into electrical "states" in bistable devices. The states of these devices can then be sent in the form of electrical pulses, at very high speeds, to the desired location where they cause other bistable devices to assume the same states as the sending devices. Thus, strictly speaking, the transmission of symbols or data involves causing a symbol at one location to generate a corresponding symbol at another point.

parallel
and
serial

There are two modes, parallel and serial, of storing and transferring digital data. In parallel operation, all the pulses comprising the computer word or message occur at the same time but on different paths or wires. In serial operation, all pulses occur on the same path, but follow each other in time.
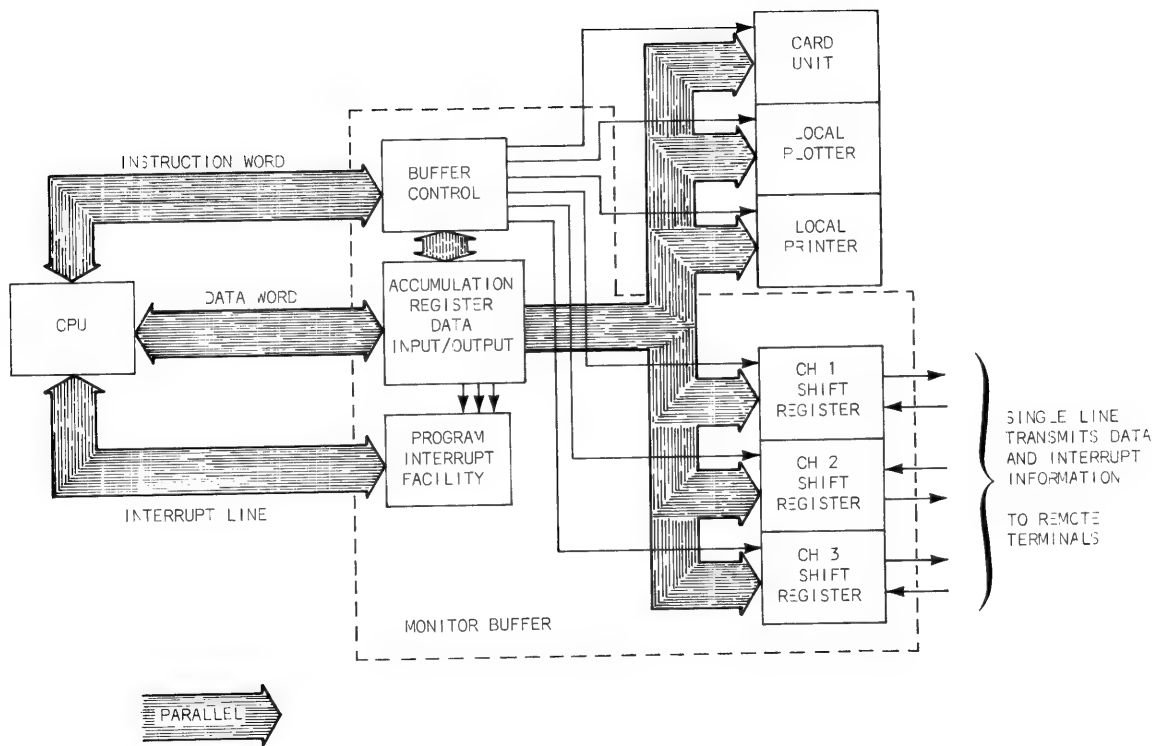
Fig. 5-1.   Monitor buffer and connective equipment.

Most large and many small digital computers operate internally in the parallel mode because it takes much less time to transmit a given amount of data, however, many wires are required between transmission points.

The telephone system is a twisted-pair hookup, but is one wire in the sense that the connection provides one signal path. A many-wired parallel telephone hookup would be extremely expensive and is generally not available. Digital data must then be transmitted over the phone line in serial form.

monitor
buffer

The monitor buffer converts the parallel digital data in the computer to serial digital data for acceptance by the phone modulator-demodulator. It also acts as a warehouse where the data can be stored until the control or "executive" program okays its release to the terminal. In the case of incoming messages, it holds them until the computer is ready to accept them. Incoming messages are in serial form and must be converted by the buffer to parallel form for use by the computer. The monitor-buffer block is as shown in Fig. 5-1. The CPU (Central Processing Unit) instruction words are transferred to and from the "monitor" in parallel form. The monitor buffer deals with all the time-sharing terminals as well as the local equipment located in the computer center. (Chapter 2.)

The monitor buffer talks to the CPU through the use of program interrupts, sometimes called "flags." For this discussion, let's assume the computer uses a 24-bit word and sends this word in parallel to the monitor buffer. 12 of the bits are *data bits* that will be forwarded by the monitor buffer to the terminal. The other 12 bits are *instruction codes* that will tell the monitor which of the terminals is to receive the data.

The monitor buffer is the in-between man and keeps an eye on the data lines from the terminal as well as waiting for instructions from the computer. When data is sent *to* the terminal it is "echoplexed" -- sent back by the terminal automatically. The monitor receives the "echo," and interrupts the computer for more data. The monitor controls the speed that the data is sent to each terminal. Of course, the monitor receives its instructions from the "executive" program.
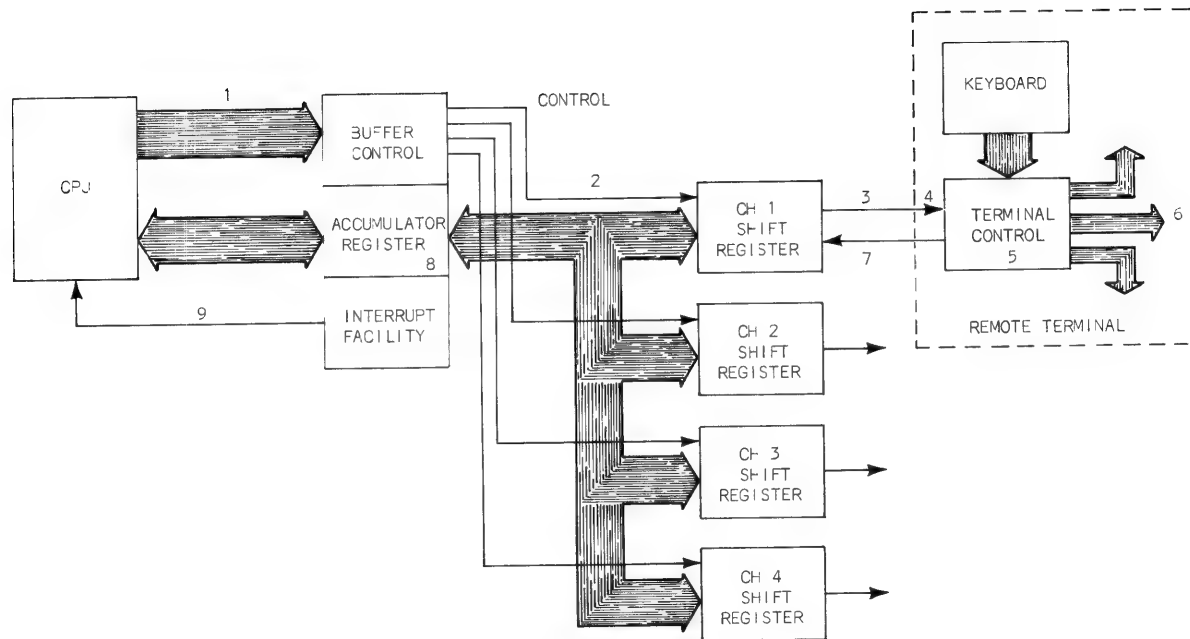
Fig. 5-2. Computer "send" operational sequence.

The executive could have different priorities for
the various time-share terminals. A company president
might want a higher priority for his terminal than
that of his company's testing lab. Or priorities
might be handled in such a way that the user specifies
his priority when he dials up the computer. The
higher the priority he requests, the faster his data
is serviced by the executive. But he also *pays more*
for a unit of computer time. The low priority user
waits longer, in some cases, but *pays less*.

Most remote terminals also have a key (often labeled
INTERRUPT) that when pressed, immediately sends a
"terminal interrupt" to the monitor buffer. This
"terminal interrupt" tells the buffer to "interrupt"
the computer with a high priority. The computer
immediately jumps to a routine to service the
interrupting terminal.

The data transfer can be understood more fully if a
transfer sequence is run through for a typical time-
share system. Individual time-share systems may
vary from this sequence due to different hardware
and software arrangements, but the exercise will
lend insight into one way a transfer could be
accomplished. Both a "computer send" and a "terminal
send" operation will be discussed. Refer to Fig. 5-2.

In a send operation from COMPUTER to TERMINAL:

1. The computer sends DATA WORD and INSTRUCTION
   WORD in parallel to the MONITOR BUFFER.

2. MONITOR BUFFER transfers DATA WORD from its
   ACCUMULATION REGISTER to its CHANNEL SHIFT
   REGISTER designated by INSTRUCTION WORD.

3. CHANNEL SHIFT REGISTER shifts information
   serially out to terminal via phone line.

4. TERMINAL CONTROL receives DATA WORD.

5. TERMINAL CONTROL locks out the terminal
   keyboard and other terminal output devices.

6. TERMINAL CONTROL routes the data or performs
   the operation designated by the DATA WORD.

7. TERMINAL CONTROL sends the DATA WORD back
   to the MONITOR BUFFER. This is known as
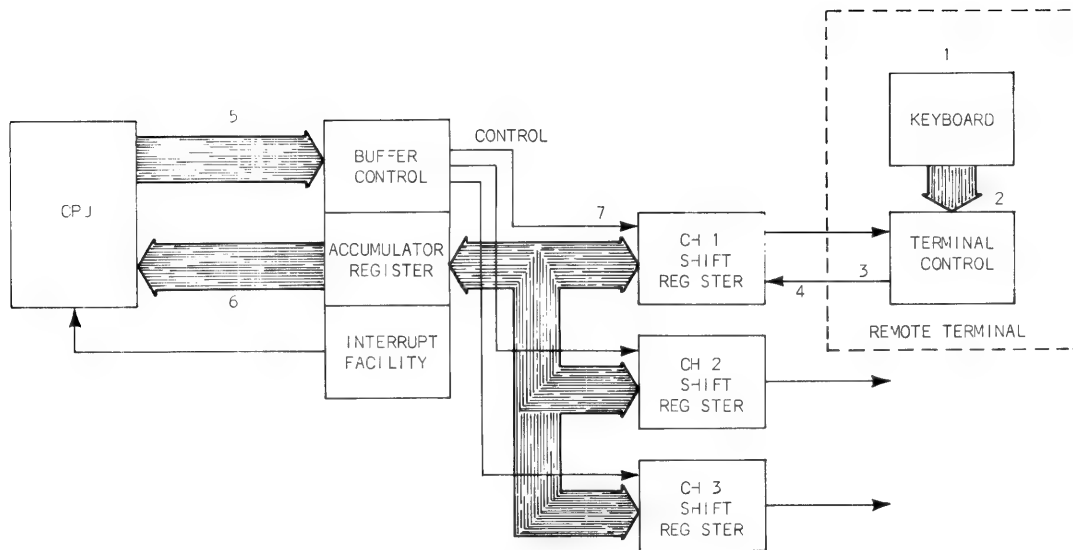   "echoplexing."

Fig. 5-3. Terminal "send" operational sequence.

8. MONITOR BUFFER receives echo.

9. MONITOR BUFFER signals computer with an "INTERRUPT" that it is ready for more DATA. The cycle can be repeated.

Refer to Fig. 5-3.

In a send operation from TERMINAL to COMPUTER:

1. Operator presses key on terminal keyboard.

2. KEYBOARD transfers DATA WORD containing ASCII CODE for that key to TERMINAL CONTROL.

3. DATA WORD is shifted by TERMINAL CONTROL out over the phone line.

4. DATA WORD is received by CHANNEL SHIFT REGISTER.

5. COMPUTER sends INSTRUCTION WORD to MONITOR BUFFER telling it to accept information from the channel. This instruction is part of a general routine in the time-share executive program and is done on a periodic basis. The channels are stepped through so rapidly that the data is accepted from the remote channels with only slight delay.

6. DATA WORD is entered into MONITOR-BUFFER ACCUMULATOR REGISTER and transferred to the COMPUTER.

7. MONITOR BUFFER echoplexes DATA WORD back to TERMINAL CONTROL.

8. TERMINAL CONTROL receives DATA WORD. It now can accept another DATA WORD from the keyboard. If ECHOED WORD contains an error compared to the original, the TERMINAL CONTROL can perform an operation (such as light a warning light) that will notify the user that an error could have been sent.

The sequences described take place very rapidly. The MONITOR BUFFER is actually transferring data in this manner to many terminals, combining send and receive operations upon command from the executive program. Computer systems can vary widely in their handling
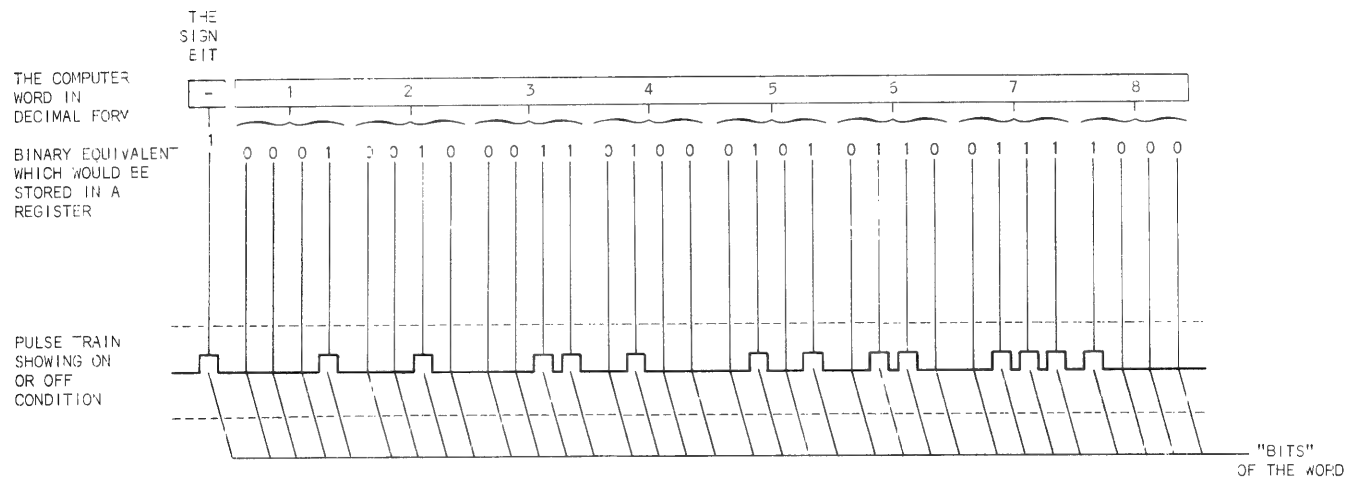
Fig. 5-4.   The computer word.

of input-output servicing of terminals. The use of a monitor buffer of some sort and a system of "flags" or interrupts is universal.

computer
word

To understand the monitor buffer further, an understanding of the computer word is necessary. The information in a computer is always in the form of binary numbers. The binary numbers could represent letters, instructions or decimal digits plus the sign. In the binary machine, this would be 20 to 40 binary digits.

The unit of data that the computer is able to handle is called one "word." See Fig. 5-4. The computer usually deals with one word at a time, selected from data-storage units called "registers." The computer register is of standard size, big enough to hold one word. The computer memory can be thought of as many one-word registers.

Each register has space for as many data states as needed to make up one word. The states are the "bits" of information. "Bits" is a contraction for *bi*nary dig*its*.

A register is built around a bistable element, which may be vacuum-tube or transistor flip-flop, magnetic cores, ferroelectric cells or any form of binary storage device. There is one bistable element for each "bit" of data in the computer word. A computer with a 30-bit word would use registers with 30 bistable elements. Each bit could be in either its "one" or "zero" state -- on or off. The *word* information would depend on the states of all the flip-flops in the register.

The computer word is often broken into information sections that the computer can handle individually. These sections are called "bytes." For instance, a computer might have a 24-bit word, but it is capable of working with the word in three 8-bit bytes.

shift
register

The monitor buffer has a one-word register of a special type known as a "shift register" for each time-share channel.
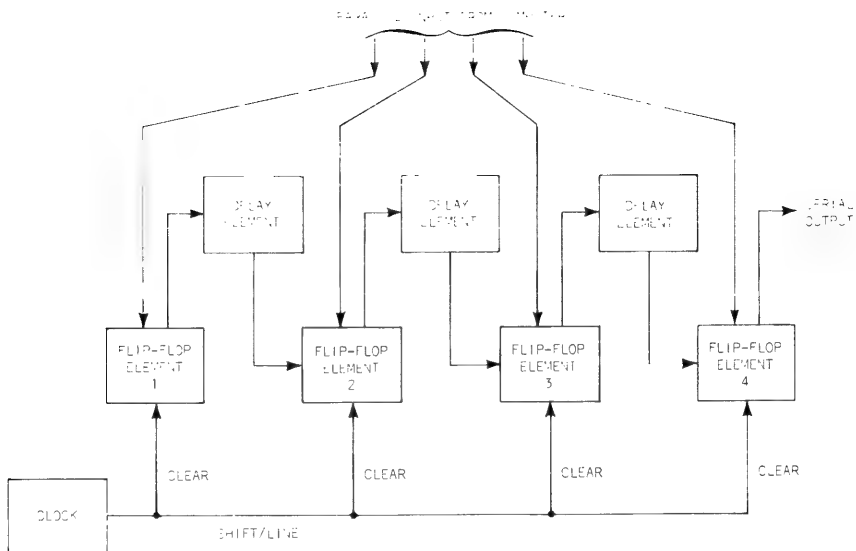
38

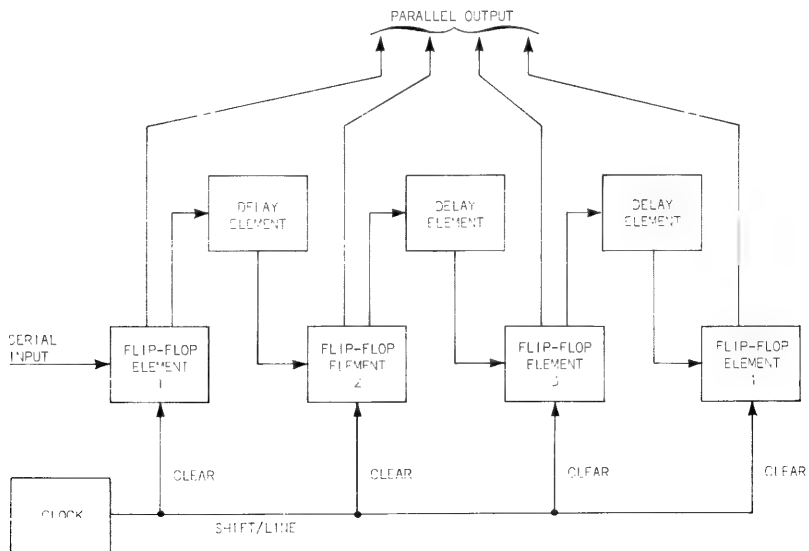Fig. 5-5.   Shift register -- parallel to serial.

Fig. 5 6.   Shift register -- serial to parallel.

As explained previously, the computer enters a "word" it wishes to transmit to the terminal in parallel form. The word is stored in the register until the computer's control unit is assured the terminal is ready to receive. When the terminal is ready, the bits are "shifted" out of the buffer register in serial form. Each stage of the shift register is built around a bistable element, together with control circuitry that upon command causes the contents of each stage element to be transferred to the next element. Information is shifted down the register by pulsing all stages simultaneously, thus causing each stored bit to advance simultaneously. The information in each stage must be stored somewhere while the following stage is being cleared. This intermediate storage is done by capacitors, by delay lines or by the turnover time of the flip-flop itself.

shift-
register
operation

In the shift register block diagram shown in Fig. 5-5, the content of each bistable element is shifted to the next higher-order element when a shift pulse is applied on the line indicated. The shift pulse acts as a *clear-and-shift* signal.

The output of element 1 is connected to the input of 2 by a delay element. The delay element retains the state of element 1 for a short time after the contents of element 1 have been cleared (shifted) and passes the state of element 1 on to element 2, which has also just been cleared. Element 2's contents are passed on to element 3 at the same time, and so on.

The output at the end of the shift register is a serial-pulse train made up of the contents of the register. The first pulse out is that of element 4 and the last that of element 1. Four clock pulses would be required to shift out the entire contents of this register. The shift pulse comes from a buffer clock. The clock is set at a rate that will allow the transmission system enough time to handle the serial bits.

The same shift register operated with input from the opposite end and located at the receiving end of the phone line could be used for serial to parallel conversion as shown in Fig. 5-6.

COMPUTER REGISTER
MSB   LSB

| C | 1 | 1 | 1 |

PARALLEL INPUT

| C | 1 | 1 | 1 |

SHIFT REGISTER

SERIAL PULSE TRAIN

0   1   1   1

TERMINAL REGISTER
MSB   LSB

| 0 | 1 | 1 | 1 |

PARALLEL OUTPUT

| 0 | 1 | 1 | 1 |

SHIFT REGISTER

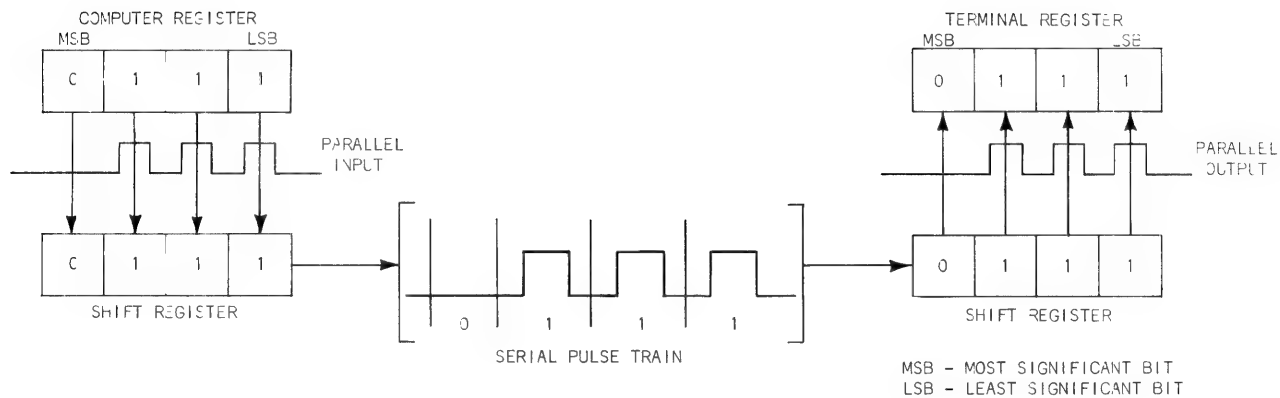MSB – MOST SIGNIFICANT BIT
LSB – LEAST SIGNIFICANT BIT

Fig. 5-7.   Parallel to serial and serial to
parallel conversion for transmission.

Using actual binary numbers and pulses, the system
would then work as in Fig. 5-7. The binary number
seven (0111) is to be sent. The pulses are entered
in the shift register in parallel from the computer.
They are shifted into serial form, transmitted and
received in serial form at the receiving shift
register. After being shifted they are again sent
to the terminal devices in parallel form.

modulation
methods

There are seven modulation methods for digital data
transmission. They are amplitude modulation (AM),
single sideband (SSB), frequency modulation (FM),
phase modulation (PM), pulse code modulation,
suppressed-carrier double sideband and single sideband
with carrier.

The considerations in choice of modulation method
are error control, equipment cost and complexity,
available bandwidth in the transmission channel,
signal-to-noise ratio of the channel, and the channel
speed capacity in bits per second. These
considerations are weighed against the application.
The choice of a data transmission method for telemetry
to a satellite might well be different than for
a phone patch between cities.

In the case of time-sharing computer systems, the
telephone appears the most practical transmission
system. Microwave and satellite systems might someday
be popular. Pulse code modulation looks attractive
because of its speed capacity in bits per second.

phase
modulation

In recent years phase modulation (PM) has become the
leading modulation method in commercially important
digital data transmission for applications like
telephone transmission, where high data-transmission
rate is a more important factor than the cost of the
equipment.

MODEM

The telephone companies rent modulator-demodulators
(MODEMS) like telephones and call them "data sets"
(Western Union) or "data phones" (AT&T). There are
a number of models available. These models are
specialized: For binary digit transmission at a
variety of data speeds; medical-information
transmission, such as electrocardiograms; very stable
repeaters for military usage; units for transmission
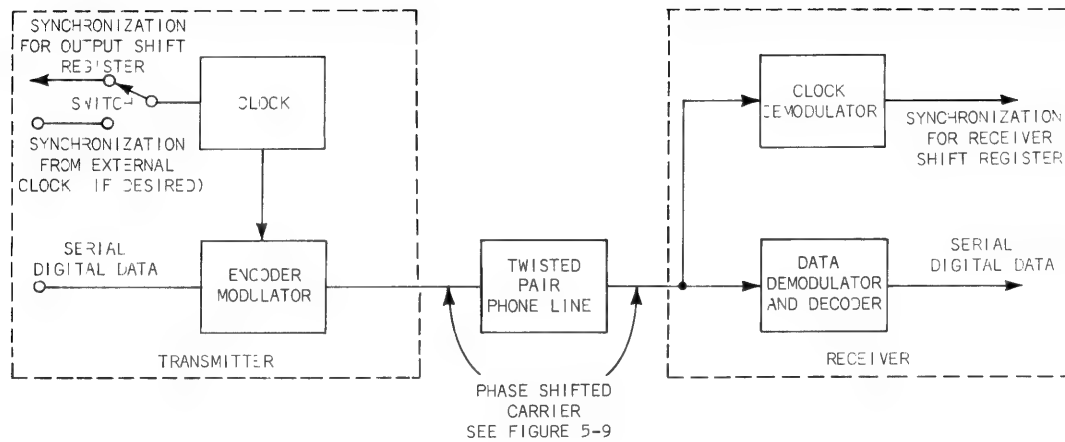of a fixed number of characters and analog facsimile
transmission.

Fig. 5-8.   MODEM block diagram.

The Model 201A and 201B data sets are the MODEMS
most often used for time-sharing. These sets, also
known as "four-phase" sets, are chosen for their
fixed high-transmission speeds and low-error rates.
They accept serial binary information from the
information source at the transmission end and
deliver it in serial form to the receiving end.
bauds   The 201A transmits 2000 bits per second (bauds) and
the 201B 2400 bauds. The 201B is designed primarily
for private line use. The 201A can be used on a
switched-voice telephone circuit or private line.
("Baud" is used for bits per second. It is named
after J.M.E. Baudot, who produced the first successful
teletype system using a five-unit code.)

These sets are designed to achieve one-bit-per-cycle-
of-bandwidth data rates. The design considerations
include the way the data is encoded as a phase
modulation (PM), the stored reference in the receiver
and the method by which continuous timing
synchronization is obtained in the receiver, directly
from the line signal. The transmission speed of
2000 or 2400 bauds is determined by a crystal
oscillator in the transmitter or by timing signals
furnished by the inputting machine. The carrier
frequency of the 201A is 1750 Hz, and the line signal
spectrum covers the range from 750 to 2750 Hz.
The 201B carrier band is from 600 to 3000 Hz with
a center at 1800 Hz.

The system block diagram is made up of a transmitter
and a receiver connected by a twisted pair phone line.
Fig. 5-8.

transmitter   The transmitter must be able to synchronize or be
synchronized with the incoming data and then operate
on the data to use it to phase modulate the MODEM
carrier. The clock, as shown in Fig. 5-8, has the
ability to free-run and synchronize the transmitter
operation. The data to be transmitted must arrive
at a rate that coincides with the transmitter's
ability to handle it. Therefore, the clock has an
output to the external equipment. This clock is
used to gate the computer or terminal-output shift
register. Should the shift register have a built-
in clock of its own, the MODEM transmitter clock
can be switched to a slave mode where it is triggered
by the external equipment. The external clock can
be run slower than the speed that the internal
clock is capable of but never faster.

dibits

The data transmission cycle starts by having information received in serial form at the transmitter from the inputting-data source. The information is examined as pairs of binary digits called "dibits." For any pair of binary digits, there are four possible dibit combinations (or codes): 00_2, 01_2, 10_2, 11_2. The phase-logic and phase-computer circuits convert these codes to four possible phase shifts of the carrier frequency. The phase shifts are odd multiples of 45° (i.e. the phase may be shifted 1,3,5 or 7 times 45°), with respect to the previous signal element's phase. This means that the phase of the carrier for a particular dibit is shifted some predetermined amount with respect to the phase just transmitted for the previous dibit. This is different from phase-modulation systems which phase shift with respect to a fixed phase reference. The data-set system makes it unnecessary to transmit *absolute* phase information.

The transmitter uses two carrier generators and a mixer. The channels alternate in supplying the line signal, with the transfer taking place gradually on each dibit. In other words, one channel will be on line during one dibit, the other during the next and so on. Phase changes are made during the time the channel is not supplying the line signal. Both channels are amplitude modulated at the dibit frequency so that they are at a minimum amplitude during channel transfer and a maximum amplitude at mid-dibit. This technique produces a line signal which changes slowly as the signal from one channel takes over from that of the other. "Splatter" of energy over the spectrum due to abrupt changes is avoided. The dibit frequency is one-half the bit frequency. (For a 2000-baud data set it's 1000 Hz.)

line signal spectrum

The output of the transmitter is then a line signal, with a spectrum consisting of a phase-shifted carrier. The phase shifts are odd multiples of 45° and contain the digital information in dibit form. The phase shifts take place at the dibit rate creating pairs of frequencies separated by exactly the dibit frequency *regardless of the data content*. See Fig. 5-9.

receiver

The receiver has two main functions: data recovery and synchronization recovery. The incoming line signal is simultaneously presented to the data demodulator and decoder and synchronization-recovery circuits. The operation of each will be discussed separately.
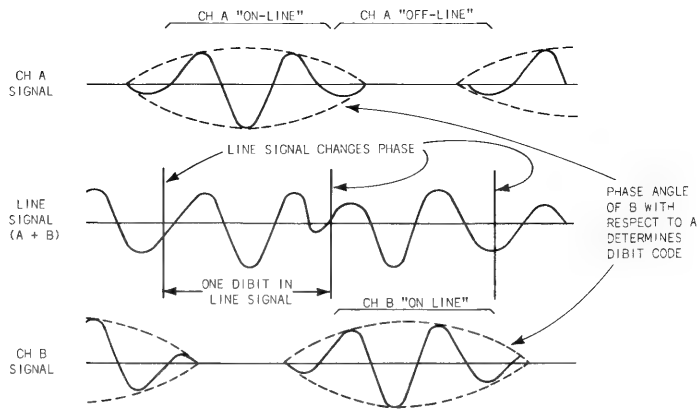
Fig. 5-9. MODEM line signal.

synchroni- zation recovery	The object of the synchronization circuitry is to recover the dibit clock frequency from the phase-modulated carrier so that the data demodulator and decoder and the user's external equipment can be set in time with the rate that the dibits were modulated onto the carrier in the transmitter. Without this timing coordination, the data could not be recovered. With this timing, the shift register at the receiving end is able to shift at the same rate as the shift register at the sending end.
	Since the line frequency spectrum always contains a pair of frequencies that are separated by the dibit frequency; that frequency may be recovered by separating the high and low sidebands by filtering, applying them to a product demodulator and tuning the demodulator output to the dibit frequency. The output is the dibit frequency used to synchronize the data-recovery portion of the receiver. The dibit clock frequency is provided for the terminal use to afford synchronization of the shift register on the receiving end.
data recovery	Data is recovered from the signal in the data demodulator and decoder. This operation is timed by the clock signal recovered in the synchronization portion of the receiver so that the output data rate matches the input data rate.

In this way, serial digital information is encoded, transmitted at voice frequencies over the phone line, received and decoded into serial digital information, and presented as an output to a shift register, computer or remote terminal at a distant point.

6

COMPUTER DISPLAY TERMINALS

Once we have the communication link between the time-sharing computer and the remote location, we need some hardware that can change the transmitted digital data into a form the operator can recognize. This hardware is the remote terminal.

The remote terminal must accept the computer word in serial form from the data set, and convert it to letters, numbers, symbols, dots or lines. A terminal could range in complexity from an indicator light array to a small analog computer driving a CRT.

The three most widely accepted remote display terminals in use are the teletypewriter, a "refreshed" cathode-ray-tube terminal and the direct-view-storage-tube terminal. We will discuss these terminals in greater detail.

teletype

Since teletype was readily available and a large communications network already established to handle teletype transmission, teletype consoles were used as the first multiple terminals. Many such terminals are still in operation, and useful for certain applications.

The most popular teletypewriter sets are Teletype Corporation's models 33 and 35. Block diagrams are shown in Figs. 6-1 and 6-2. This system is basically a mechnical-to-electrical conversion for transmitting, electrical-to-mechanical for receiving. The display is, of course, the teletypewriter's paper.

Teletypewriters transmit and receive messages by a binary code. The characters making up the messages are represented by prearranged combination of binary intelligence levels. For transmission, the intelligence is placed in electrical form, referred to as the "start-stop" signaling code. The binary bits are applied sequentially (serially) to the transmission line as current or no-current time intervals. The time interval where current flows is called "marking," and the no-current interval "spacing."

The teletype operates on a modification of the American Standard Code for Information Interchange (ASCII). A word in ASCII code has seven bits. There are thus, 2^7 (128) different words in the code. (See Chapter 7, Fig. 7-1.) The teletype uses 64 words for printing characters. The rest are devoted to control or are unassigned. The teletype machine works on *eight* code bits, with its stop element two units of mark time. The pattern is then as seen in Fig. 6-3, eight code units with an 11-unit transmission time. The eighth bit is an "even-parity" unit to provide error detection.

In a "send" operation, the operator depresses a key. The codebar mechanism is mechanically positioned to "make" eight contacts. The output from these contacts is fed in parallel form to the distributor. The distributor is a mechanical device that converts the parallel information to serial "mark" or "space" outputs to the transmission line. (Fig. 6-1.)

In a "receive" operation, the received "marks" or "spaces" pull down magnets in the selector-magnet driver. The driver output is fed to a selector mechanism, that mechanically positions the codebar mechanism. The codebar mechanism activates the proper key-function mechanism, printing the desired character. (Fig. 6-2.)
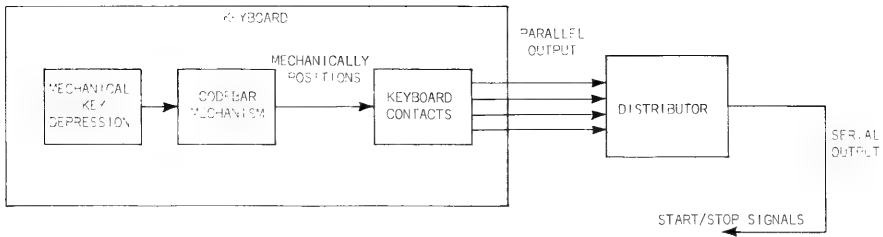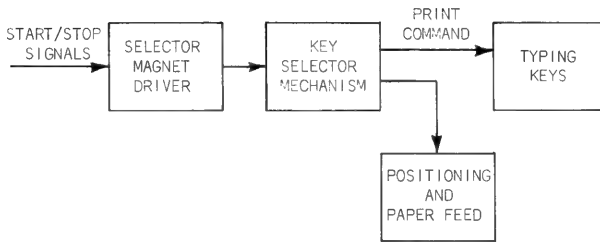
Fig. 6-1. Teletype -- send.
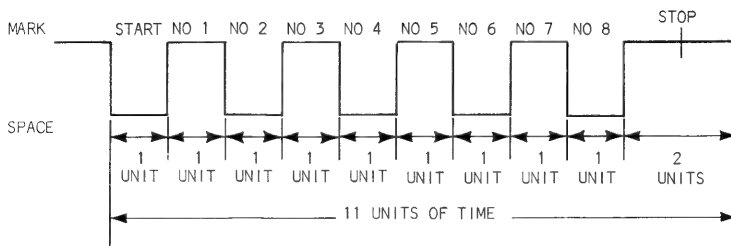

Fig. 6-2. Teletype -- receive.

Fig. 6-3. Current waveform for teletype letter
"U" showing B-level code with 11-unit
transmission pattern.

The chief advantage of the teletype system is the availability of an already existing communications network (ease of installation) and its hard-copy output. The remote terminal user who wants a permanent record of transmitted data need go no farther with the teletypewriter printing on paper. Also, by punching a paper tape off-line with the teletypewriter, and then inputting that information from the tape, less transmission time is used than when trying to compose on-line by hand typing.

The first users of the time-share teletype terminal soon realized that this terminal utilized much less capability than was available from the computer. There were four major limitations:

1. Fixed format. The teletype must print as a typewriter (left to right and top to bottom) and includes only the standard teletype symbols.

2. Slow print rate. The maximum speed is 200 words per minute. This is uneconomical in that it does not use the full capability in bauds of the telephone-line bandwidth (5000 words per minute for a 2400-baud line), and disconcerting to the user who must wait for feedback, that is not as immediate as desirable.

3. No graphics capability. Again the fixed format prohibits graphical output except for time-consuming, limited-quality plots, utilizing spatial arrangement of alphamerics.

4. Noise, and short mean-time-to-failure of mechanical devices.

cathode-ray-tube terminal
The need and desirability of graphics as well as of overcoming the other limitations of the teletype system stimulated the development of terminals using cathode-ray tubes as the display media.

The cathode-ray tube has a persistence time from 0.05 to 95 ms, depending on the phosphor. The longer-persistence phosphors are marginal as display devices; they are distracting to the viewer, due to the phosphor color change from initial phosphorescence to fluorescence. They also have short life. The

solution was, then, to use a medium-persistence phosphor and to rewrite or "refresh" the display at a rate that prevents flicker. Thirty times per second appears to be the lower practical limit for medium-persistence phosphors. For extended phosphor life, some refreshed units have gone to P31 phosphor (very long life). Because of P31's short persistence, these units require 60-Hz refreshing. The "lower limit" on refreshing is the point where the flicker becomes noticeable to the user. At first glance, a small amount of flicker does not seem objectionable, but flicker rapidly causes eye fatigue. After a while, the user finds it objectionable. Most systems do not restrict the flicker rate to a fixed lower limit; where the system does not function below that limit. Instead, they allow the refresh rate to vary with the amount of information that must be written. At high information densities, the refresh cycle may take so long that the display blinks. Another drawback of the variable refresh rate is that, at certain frequencies, it "beats" with fluorescent lamps becoming more noticeable and objectionable.

Recall from Chapter 5 that the telephone communications link has a maximum data rate of 2400 bauds. That is equivalent to 0.41 ms per bit. If the user wishes to have a usable repertory of 64 symbols, at least six bits per symbol must be transmitted (2^6). Each symbol would then require 2.5 ms. With a 30-Hz refresh rate (once every 33.3 ms) the maximum number of symbols the computer could provide at a flicker-free rate via phone line would be 13. Such a small amount of data proved inadequate except for very limited systems that use a very small number of symbols. For most systems, a local memory was added to the terminal to provide refreshing. The telephone transmission rate now applies to loading the local memory; but once loaded, the memory can repeatedly drive (i.e. refresh) the CRT at an increased rate.

Refreshed CRT terminals are usually of two distinct types; low-cost fixed-format, limited-capability or sophisticated, high-priced, multi-featured. The sharp division between the low-cost system and the sophisticated one is brought on by the rapidly mounting costs of memory and circuitry as the data to be displayed increases. Both types are discussed in detail.

nongraphic
refreshed-
CRT
terminal

The low-cost, fixed-format, limited-symbol system displays a fixed number of lines and characters per line -- perhaps 10 to 20 lines and 50 characters per line, for a maximum of 1000 characters. Graphics are not available. The CRT size is usually an 11" diagonal.

The refresh memory is a relatively inexpensive acoustic or magnetostrictive delay line. These delay lines are limited to about 6000 to 8000 bits if a 30-Hz refresh rate is maintained. Because even with a limited format it takes about 6 bits per character, this system is limited to the 1000 to 1300 characters mentioned above. This system finds application in a number of fields, such as banking, stock-market quotations and record-keeping.

A block diagram of a low-cost refreshed system is shown in Fig. 6-4. The computer information is received, converted to characters by the character generator and entered into the memory at the same time. The characters are displayed and redisplayed each time the memory cycles.

The display controller synchronizes the operation of the terminal. It interprets the code bits from the computer word and routes the computer information to activate the indicated remote terminal equipment.

The controller also interprets inputs from both the keyboard and computer, such as positioning, spacing, line feed, backspace, rub-out, erase and cursor.
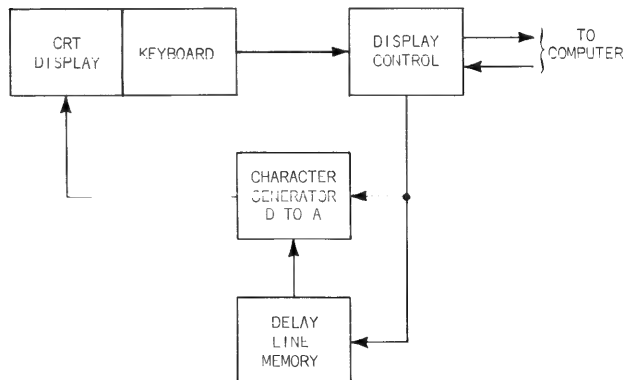

Fig. 6-4. Fixed-format limited symbol refreshed CRT terminal.

delay-line
memory

In a delay-line memory system, the pulses from the
computer in serial bit form are stored by repeatedly
propagating them in a closed loop from a transmitter
to receiver via a stationary medium. The loop is
closed by connecting the receiver to the transmitter
via gating and shaping circuits. Since it takes
time to propagate the energy, the pulses in transit
can be considered stored in the delay line. Delay-
line storage has been done by electrical, acoustic,
electromagnetic, piezoelectric or magnetostrictive
phenomena. By repetitively re-introducing the
signals into the delay medium, in synchronism with
a time-reference pulse, the temporal position of a
data bit can always be specified. At the speed
required, the delay medium is being used beyond its
limited bandwidth, causing distortion and attenuation
of the circulation signal. Therefore, circuitry is
provided to amplify and reshape the pulses.

The operation of a delay-line storage is as follows
(Fig. 6-5): Temporal serial binary information is
received from the computer and gated into one end of
the delay element. As a result, the temporal serial
information becomes a spatial serial pattern as the
data passes through the line. Recirculation of data
is provided by closing the loop from the end of the
delay line back to the beginning by means of
transducer, amplifiers, shaping and synchronizing
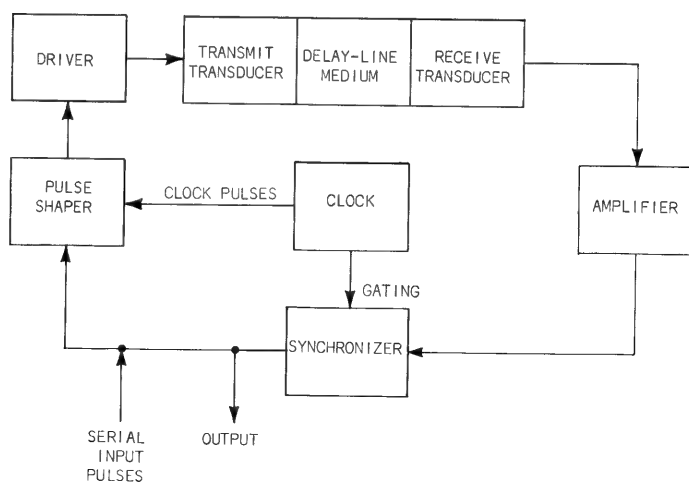circuits.

Fig. 6-5. Delay-line memory.

The amount of information that can be stored depends
on the delay medium and its length. Acoustic and
magnetostrictive delays appear to be the most
popular. Typical characteristics of magnetostrictive
delay lines are minimum bit rates of 100 kHz and
maximum to 1 MHz. Typical lengths (8 ms) allow
storage of up to 8000 bits.

sophisticated In the sophisticated refreshed system, performance
refreshed- is the criterion and cost is somewhat secondary.
CRT When the quantity of alphamerics increases or
terminal graphics are desired, the cost of the refreshed
system is substantially increased. Delay lines
can no longer be used as memory. For instance, a
fixed-format display of 50 by 50 symbols utilizing 6
bits per character would require 15,000 bits of
memory (2500 symbols x 6 bits per symbol). A
random format would require an additional 12 bits of
positioning data for each symbol: 6 bits of X
position and 6 for Y. Thus, a random-format 50-by-50
symbol display would demand a local memory of 45,000
bits (2500 symbols x 18 bits per symbol).

The number of bits required to do graphics by point
plot or vector mode depends on the complexity of the
plot. A plot such as Fig. 6-6 required 80,000 bits
of memory. An even greater expense is necessary in
the high-speed circuitry necessary to cycle the memory
and present the characters or graphics at a flicker-
free rate. Assuming a 30-Hz refresh rate, each
refresh must be done in 33.3 ms. The 50 by 50 symbol
display would require refreshing at about 13 μs per
symbol. If this symbol were made by dot generation
from a 7 x 9 dot matrix, each dot would have to be
written in 200 ns. This speed requirement makes dot
generation impractical.

To be able to write characters at this speed, the
large-capacity refreshed system was forced to go to
"stroke" generators. The stroke generator is actually
a small analog computer that outputs X and Y
deflection voltages, producing characters by line
segments or "strokes." High-speed stroke generators
are a practical, although expensive, way to achieve
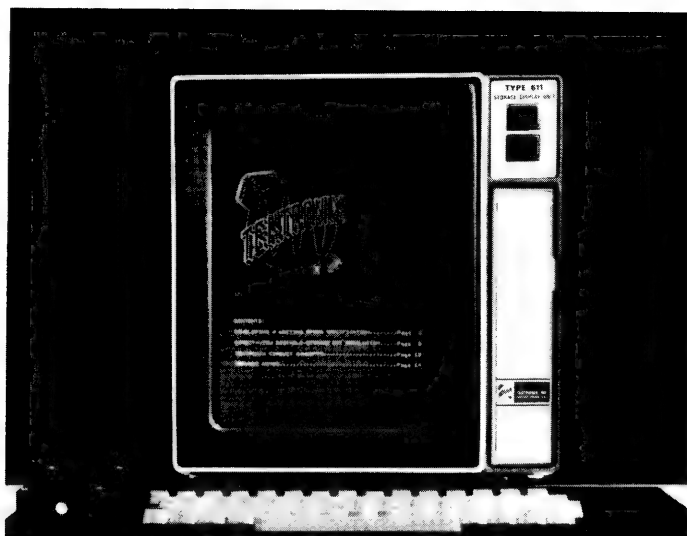a large-capacity refreshed display.

Fig. 6-6. Tektronix Type 611 Storage Display Unit.

light pens

A natural result of having the local memory was the
use of "light pens" and a "selective erase" mode.
The light pen is a light-sensing device which could
be used to locate a point on the CRT display and
signal the computer to perform an operation at that
point. Some light-pen systems provide an error-
sensing feedback mode that allows the CRT beam to
follow the pen movements, simulating "writing" on
the CRT with the pen. The light pen is discussed
in detail in Chapter 7.

selective
erase

Since the display information is stored in a *local*
memory, the memory can be readily modified, thereby
changing the display. Selective erase systems can
erase a character from local memory without affecting
the rest of the display. A new character may be
entered in its place. The cost of additional
circuitry required for selective erase is small
compared to the total refreshed-system cost, so most
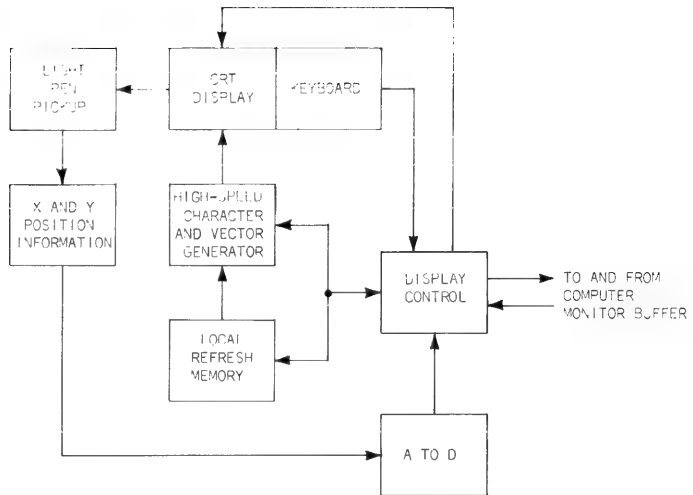refreshed systems provide this capability.

Fig. 6-7. Sophisticated refreshed CRT system block.

memory
systems

The sophisticated system operates much like the simple system. Fig. 6-7. The memory system must be of greater storage capacity and therefore is not a delay-line type.

Drums, discs, cores and semiconductors are used for memories. The big question is cost. Most users rank semiconductors, discs, cores and delay lines in that order of descending cost per bit.

Each method has its own problems. Delay lines have inherent jitter that can cause a wavy display. Discs have certain mechanical difficulties, with belts and pulleys. Cores are relatively free from both of these problems. The circuitry to input and output from memory, and for character and vector generation must be of high speed to accomplish rapid refresh of a high-density display.

memory
character-
istics

Memory systems are usually distinguished by operating speed, volatility, erasability and access time. Besides these characteristics, the cost per bit and physical size are of particular importance to the terminal user.

"Operating speed" refers to the rate at which data is transferred into and out of the storage system. Once access is gained to a location in memory, the rate at which the information is read out will depend on the type of storage.

"Volatility" indicates whether power must be applied to retain stored information. A volatile storage system requires power, while nonvolatile does not.

"Erasability" is the ability to be erased. Punched cards are not erasable -- electrical devices such as magnetic cores, drums and transistors are.

"Access" time is the time required to gain entrance to an item in storage. In a computer it is of particular importance because it limits the overall computation time.

echoplexing
In some terminals the information is sent from the keyboard to the computer, then computer to the display, rather than directly from keyboard to display. This is done to check that the computer has received the correct information, and is called "echoplexing." In the teletype industry the same phenomena is called "half duplexing."

direct-view bistable-storage-CRT terminal
The restricted format and nonexistent graphics capability of the low-cost refreshed system and the complexity and cost of the sophisticated refreshed system pointed out the inherent advantages of utilizing the direct-view bistable storage tube (DVBST) as a low-cost, high-information-density remote computer display.

The ability of the storage target to retain the stored image eliminated flicker and the need for the local memory and associated refreshing circuitry. Flicker is a very important human-fatigue problem with long-period viewers. Character and vector-generation circuitry was simplified because the speed requirements for generating symbols at refresh speeds was removed. Basically, simple dot generation could replace "stroke" generation. The display controller performs the same function in the bistable display system as in the refresh system.
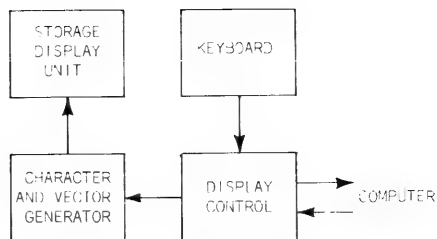
Fig. 6-8. Direct-view bistable-storage-tube
terminal.

The direct-view bistable storage-tube display terminal
has a character generator and usually a vector
generator to convert the serial digital data to
analog. The terminal also contains a storage CRT
display, display controller, keyboard and other
devices to output back to the computer through an
analog-to-digital converter. (Fig. 6-8.)

The principal advantages of the direct-view bistable
storage-tube terminal are its reasonable cost, good
resolution, no flicker and ability to display
both alphamerics and graphics in a wide range of
formats.

The character and vector generators used with direct-
view bistable storage-tube terminals are discussed
in Chapter 7; the display itself in Chapters 9 and 10.

7

TERMINAL OUTPUT DEVICES

Operator convenience is of primary importance to the terminal user. The ability to communicate with the computer with minimum fatigue and maximum flexibility is the criterion for a practical time-share terminal.

We have seen how the display can aid or obstruct this objective. The teletype, for instance, in many cases is inadequate because of its noise, slow speed and inability to communicate through graphics. The direct-view bistable storage tube has a low fatigue factor in its elimination of flicker.

Although the display is probably the most important part of any terminal, the auxillary features provided by the terminal add greatly to the man-machine interface.

These features provide the ability to respond to data on the display, to enter both graphics and alphamerics to the computer and to the display, and to alter data on the display and in the computer. In general, they let the user react and respond conveniently to the computer's outputs.

Ideally, the user would like to have a microphone and pointer. Words spoken into the microphone could tell the computer what to do. The pointer could let the computer see where a change should be made -- or might even draw a picture to be stored in the computer. (This assumes that the computer can see and hear like another man; today it cannot. Other indicating and communicating devices are needed.)

keyboard

Most cathode-ray-tube terminals use a typewriter-type keyboard to send word messages to the terminal and to write the same words on the display. The more sophisticated terminal also provides a graphic means of indicating points, or drawing on the CRT and feeding the data to the computer. The most common terminal graphic output devices are the light pen, X-Y tablet and X-Y "mouse."

Standard Code

b4 b3 b2 b1	COLUMN	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
0 0 0 0	0	NUL	DLE	SP	0	@	P	`	p
0 0 0 1	1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	2	STX	DC2	"	2	B	R	b	r
0 0 1 1	3	ETX	DC3	#	3	C	S	c	s
0 1 0 0	4	EOT	DC4	$	4	D	T	d	t
0 1 0 1	5	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	6	ACK	SYN	&	6	F	V	f	v
0 1 1 1	7	BEL	ETB	'	7	G	W	g	w
1 0 0 0	8	BS	CAN	(	8	H	X	h	x
1 0 0 1	9	HT	EM	)	9	I	Y	i	y
1 0 1 0	10	LF	SUB	*	:	J	Z	j	z
1 0 1 1	11	VT	ESC	+	;	K	[	k	{
1 1 0 0	12	FF	FS	,	<	L	\	l	|
1 1 0 1	13	CR	GS	-	=	M	]	m	}
1 1 1 0	14	SO	RS	.	>	N	^	n	~
1 1 1 1	15	SI	US	/	?	O	_	o	DEL

Legend

Control Characters

NUL	Null
SOH	Start of Heading (CC)
STX	Start of Text (CC)
ETX	End of Text (CC)
EOT	End of Transmission (CC)
ENQ	Enquiry (CC)
ACK	Acknowledge (CC)
BEL	Bell (audible or attention signal)
BS	Backspace (FE)
HT	Horizontal Tabulation (punched card skip) (FE)
LF	Line Feed (FE)
VT	Vertical Tabulation (FE)
FF	Form Feed (FE)
CR	Carriage Return (FE)
SO	Shift Out
SI	Shift In
DLE	Data Link Escape (CC)
DC1	Device Control 1
DC2	Device Control 2
DC3	Device Control 3
DC4	Device Control 4 (Stop)
NAK	Negative Acknowledge (CC)
SYN	Synchronous Idle (CC)
ETB	End of Transmission Block (CC)
CAN	Cancel
EM	End of Medium
SUB	Substitute
ESC	Escape
FS	File Separator (IS)
GS	Group Separator (IS)
RS	Record Separator (IS)
US	Unit Separator (IS)
DEL	Delete[1]

Graphic Characters

Column/Row	Symbol	Name
2/0	SP	Space (Normally Non-Printing)
2/1	!	Exclamation Point
2/2	"	Quotation Marks (Diaeresis[2])
2/3	#	Number Sign[3,4]
2/4	$	Dollar Sign
2/5	%	Percent
2/6	&	Ampersand
2/7	'	Apostrophe (Closing Single Quotation Mark; Acute Accent[2])
2/8	(	Opening Parenthesis
2/9	)	Closing Parenthesis
2/10	*	Asterisk
2/11	+	Plus
2/12	,	Comma (Cedilla[2])
2/13	-	Hyphen (Minus)
2/14	.	Period (Decimal Point)
2/15	/	Slant
3/10	:	Colon
3/11	;	Semicolon
3/12	<	Less Than
3/13	=	Equals
3/14	>	Greater Than
3/15	?	Question Mark
4/0	@	Commercial At[3]
5/11	[	Opening Bracket[3]
5/12	\	Reverse Slant[3]
5/13	]	Closing Bracket[3]
5/14	^	Circumflex[3,4]
5/15	_	Underline
6/0	`	Grave Accent[2,3] (Opening Single Quotation Mark)
7/11	{	Opening Brace[3]
7/12	|	Vertical Line[3]
7/13	}	Closing Brace[3]
7/14	~	Overline[3] (Tilde[3]; General Accent[2])

NOTE: (CC) Communication Control
    (FE) Format Effector
    (IS) Information Separator
[1] In the strict sense, DEL is not a control character. (See 5.2.)

[2] The use of the symbols in 2/2, 2/7, 2/12, 5/14, 6/0, and 7/14 as diacritical marks is described in Appendix A, A5.2.
[3] These characters should not be used in international interchange without determining that there is agreement between sender and recipient. (See Appendix B4.)
[4] In applications where there is no requirement for the symbol #, the symbol £ may be used in position 2/3.

Fig. 7-1. USA Standard Code for Information Interchange.

Simply, the keyboard provides a means of converting the mechanical action of key depression into an electrical digital code, to be transmitted to the display character generator and computer. Keyboard conversion can be mechanical-to-electrical, electromechanical-to-electrical or photoelectric-to-electrical.

The mechanical keyboard uses a ccdebar mechanism similar to the console typewriter (Chapter 2). The electromechanical keyboard operates through relay contacts or an RF field. The RF field has a Faraday shield around it. The shield has holes in it. When a key is depressed, certain holes in the shield are covered. A detector senses the field changes and transmits the proper character code. Other electromechanical keyboards use reed switches. The photoelectric keyboard uses the same shield principle, except that light is used to project through the holes. Key depression masks certain holes, which are detected by a photocell pickup. The character is encoded and transmitted. The specific code for the characters can be chosen by the manufacturer. IBM has a code that it has set up for use in all the equipment it makes.

ASCII
code

Many keyboard manufacturers are standardizing on a code that has been developed by a subcommittee of the American Standards Association. The code is known as the "American Standard Code for Information Interchange." (ASCII) The code has not been formally approved by the American Standards Association, but it is anticipated that it will be -- with perhaps minor modifications. See Fig. 7-1 for the ASCII code. ASCII code uses seven bits plus a parity bit.

The ASCII code has 96 characters and symbols. The total code has 2^7, or 128, combinations. The 32 extra codes are used for control functions such as carriage return, spacing, line feed, up/down positioning and backspace.

With a typewriter typing on paper, the user can tell where the next character will be typed by the mechanical position of the carriage. With a CRT display, there *is* *no* mechanical relationship. For

cursor   the user to know where he is on the display, a
cursor is usually provided. In the refreshed display,
the cursor is refreshed right along with the display.
With the direct-view bistable storage tube, such a
cursor would, of course, be stored in the display,
making it unusable. The direct-view bistable tube
makes use of a "write-through" cursor that does not
store, but is visible to the user.

write
through   "Write-through" is a natural phenomenon of the
bistable storage CRT.

The characteristics of the bistable target are such
that a target element in the unwritten (nonstored)
state requires a finite amount of energy to drive it
to the upper stable (stored) state. This "switching"
of the target phosphor element is accomplished by
bombarding it with electrons from the writing gun.
These high-energy electrons cause secondary emission
from the target element, with more electrons being
knocked out of the element than are arriving in the
writing gun beam. The net result is a positive
charging (loss of electrons) of the bombarded target
element. If the bombardment continues long enough,
the target element will charge above the crossover
(switching) point, and will go to an upper "written"
state. Thus, the unwritten target element is
"written" by the writing gun beam being concentrated
on it for some finite length of time.

If, however, the writing gun beam is held on the
phosphor for *less* time than that required for the
phosphor element to reach crossover, the phosphor
element will fluoresce while the beam is on and
phosphoresce to decay, but the element will *not be
permanently* stored. If the beam is writing over and
over at the same point, there must be enough time
between "writings" to allow the target element to be
discharged by the flood guns from its partially
charged state. Otherwise, the target element will
have a charge accumulation that will eventually cause
it to store.

Write-through is accomplished by using the target in
this area -- where the writing beam makes it *glow*,
but it does *not store*.

Write-through circuitry is built into some storage display units. The units provide reduced CRT-beam intensity and rotate the beam in a small cursor circle. The beam intensity and speed are such that it is visible to the user, but stays below the storage threshold and does not store.

Write-through can also be provided by the terminal character-generation equipment. If the storage CRT dot-writing time (i.e. time to store) is 20 µs, then a write-through rectangle can be provided by repetitively stepping through the character generator 7 x 9 matrix, and gating on the Z axis for 1 µs per dot, instead of 20 µs per dot. The reduced Z-axis on-time will not allow the beam to store. This method will provide write-through for any direct-view bistable storage-tube display unit, even though it is not provided internally. The cursor need not be generated using the full 7 x 9 dot matrix. A part of the matrix could be used, making the cursor in the form of a circle, arrow, etc.

Theoretically, write-through could be used to write information other than just a cursor. To be viewable, the information would have to be refreshed at a rate that allowed a bright enough trace for viewing, yet did not remain in one spot long enough to store. In most cases, write-through would be unsatisfactory for information that was much more complex than the cursor already described.

special
control
functions
The 32 codes not used for the 96 ASCII character set are used for control functions. Control functions are usually initiated by depressing a control key along with a standard character key.

The common control functions are those found on a teletypewriter: Space, line feed, backspace, carriage return, vertical position -- (up and down), horizontal position, etc. All the ASCII control characters are shown in Fig. 7-1.

Oftentimes special controls are desirable, such as slanting the writing coming from the computer, making it easily identified.
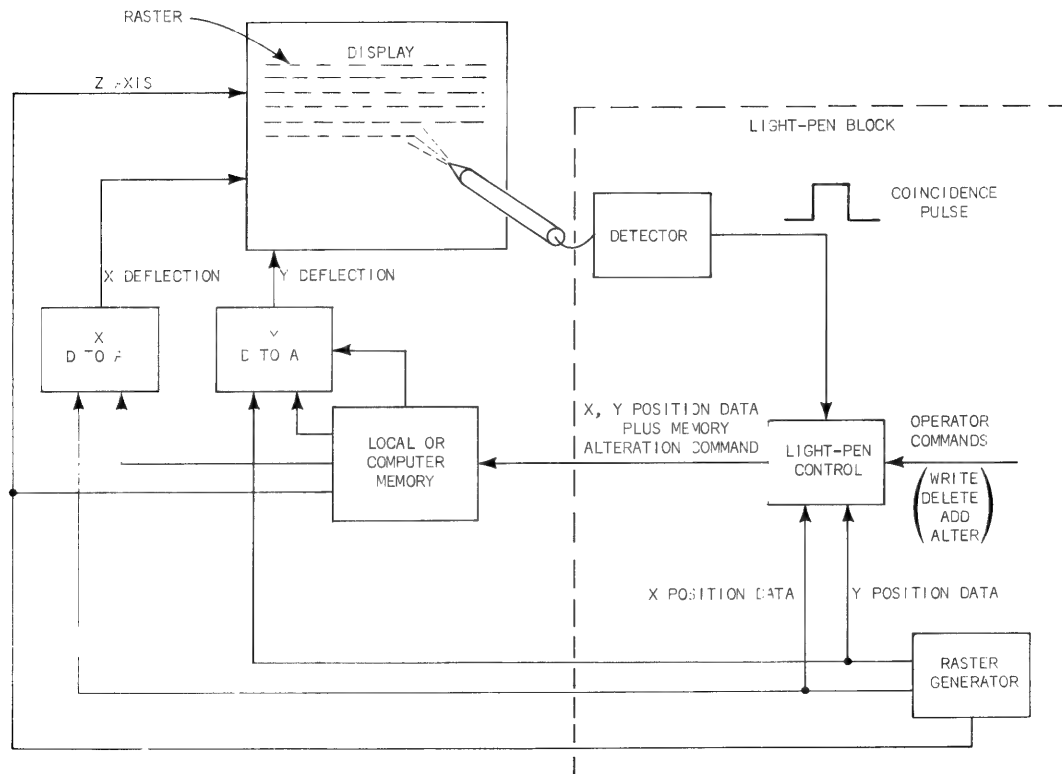
Fig. 7-2. Light-pen block diagram.

Besides being able to communicate with words, some users of a graphics terminal want to answer the computer graphically. This usually is done by light pen, graphics tablet or "mouse."

The graphical output device is a means of locating a position on the display and signaling the computer to perform an operation at that point. The outputs from the graphical device are usually X and Y analog voltages representing the display position. These voltages must be converted to digital voltages to be transmitted to the computer. The analog-to-digital converter, which performs this function, is discussed in Chapter 8.

light pen The light pen is one of the most popular means of graphically entering information on the display and in the computer. A block diagram of light-pen operation is seen in Fig. 7-2. The memory shown in the figure could be the local terminal memory, in the case of a refreshed terminal, or the computer memory for a storage terminal.

The pen itself does not "write." It is actually a photosensitive pickup that senses light changes. When aimed at the CRT display, it will generate a pulse output every time the CRT beam passes under the photocell aperture. Some pens actually emit a small circle of light, so that the user can tell where the pen is aimed on the display.

The light-pen circuitry generates a raster that deflects the CRT beam. When the raster passes under the light-pen detector, a pulse is sent back to the light-pen control. With a bistable storage monitor, the raster would be in the "write-through" mode. The control section of the pen has inputs from the X and Y raster generator and the light-pen detector. When a detector pulse is sensed, the control sends X and Y position information to the local memory or the computer memory. At the request of the operator, the control section can also tell the memory to change the display at the point of light-pen pickup. Special computer programs written for light-pen use, allow the operator to give the computer special instruction using the pen positional values -- as end points of lines when in a drawing mode, as a pointer in locating one of a number of displayed objects, and so on. Information under the pen might be added, deleted or altered.
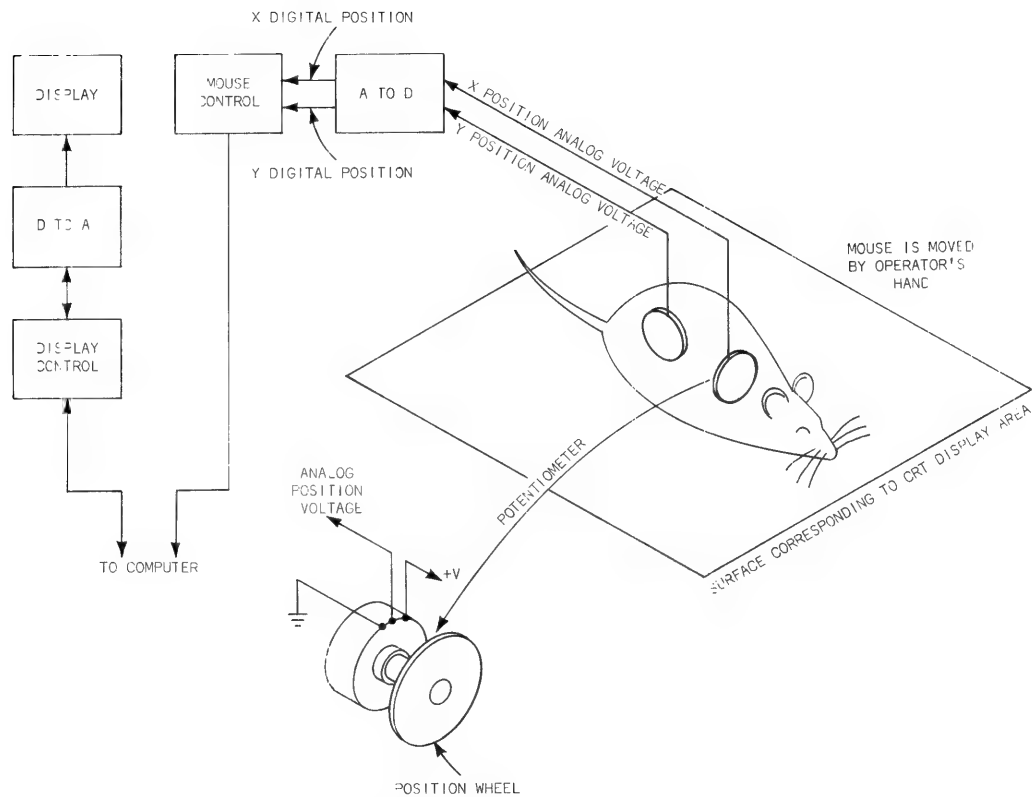
Fig. 7-3. X-Y mouse.

Many pens have a writing mode that can be called up
by the operator. In the writing mode, the control
senses the pen location and signals the memory of the
X and Y position. The memory then unblanks the Z
axis at that position. As the pen moves about the
display, the beam is turned on wherever the pen points,
a line is written wherever the pen has been. The pen
appears to "write." Some systems are sophisticated
enough to transform hand-drawn lines into precise
straight lines, upon command.

Another means of graphical output is to move a
"write-through cursor" over the stored picture, yet
not store the image of the cursor. The cursor pattern
is generated locally and its display position is
controlled by a hand-held box that is moved about on a
surface. See Fig. 7-3. The surface could be provided
as part of the terminal. The box, called a "mouse" by
its developers at Stanford Research Institute, has two
potentiometers mounted at right angles to each other.
Wheels attached to the potentiometers contact a
surface and resolve the motion of the "mouse" into two
orthogonal components that are fed as analog voltages
to the display. Thus the cursor on the screen follows
the motion of the "mouse."

X-Y
mouse

The operator can push a button requesting that the
analog position voltages be converted to digital X
and Y signals of the cursor's position and then
transmitted, along with the desired display-
modification information, to the computer. The
operator's program can then interpret the commands
as with the light pen.

X-Y
tablet

The X-Y tablet is another means of sending graphical
information from the terminal to the display and
computer.

Tablets have been constructed a number of ways:
Using a resistive contact surface; or using three
layers, consisting of a voltage grid sandwiched
between two voltage gradient conductors, or a
combination or variation of these methods.

The basic idea is to have the pressure from pen
contact send out X and Y analog position voltages.

A tablet made of a resistive surface is seen in
Fig. 7-4. The surface material is such that it acts
as a voltage divider in both the X and Y directions.
A voltage is placed on the surface by the point of
the "pen." Depending on the pen location, a
differential signal is picked up by each of the X
and Y differential amplifiers. For example, if the
pen were equidistant from the upper and lower Y
terminals, the Y differential amplifier would see
the same applied pen voltage on its plus and minus
inputs, and would output an analog voltage indicating
the pen was centered. Likewise on X. A linear
conductive contact strip is used along each edge to
eliminate pickup errors in the corners called
"fringing."

An example of the "layer" technique is shown in
Fig. 7-5. A voltage grid is placed between two
conductors. The voltage on each grid line is
different, depending on its position in the grid.
The lines, when observed collectively, form an X
and Y gradient. Pen pressure forces the upper and
lower conductors to touch the grid at one point.
The upper conductor picks up an X voltage, the lower
a Y voltage from the grid. The analog X-Y voltages
thus attained reflect the pen's coordinates.

The X-Y position voltages are digitized and used as
with the light pen and "mouse."

Undoubtedly, there are other means of terminal
graphical output that have not been mentioned here,
however, the principle of operation will always be
the same. Graphical output must include some means
of indicating the X and Y coordinates of a position
on the display, with the ability to operate on the
display at that position. With this principle in
mind, it should be easy to understand most terminal
graphical output devices.

selective
erase

Selective erase is the ability to use the keyboard
terminal or graphical output device to pick any
character, line or area on the display and delete it
from the picture.

It is usually included with sophisticated refreshed
systems because it can be had for little extra cost.
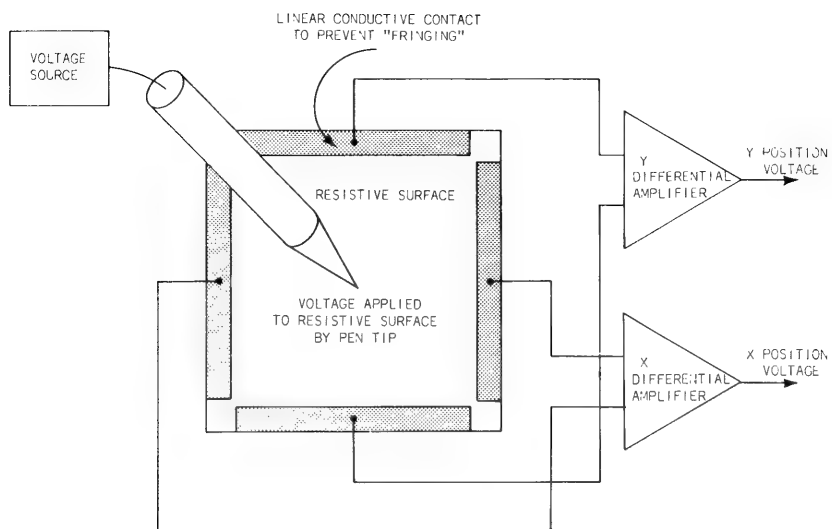The keyboard simply modifies the local memory, which

Fig. 7-4. X-Y tablet using resistive surface
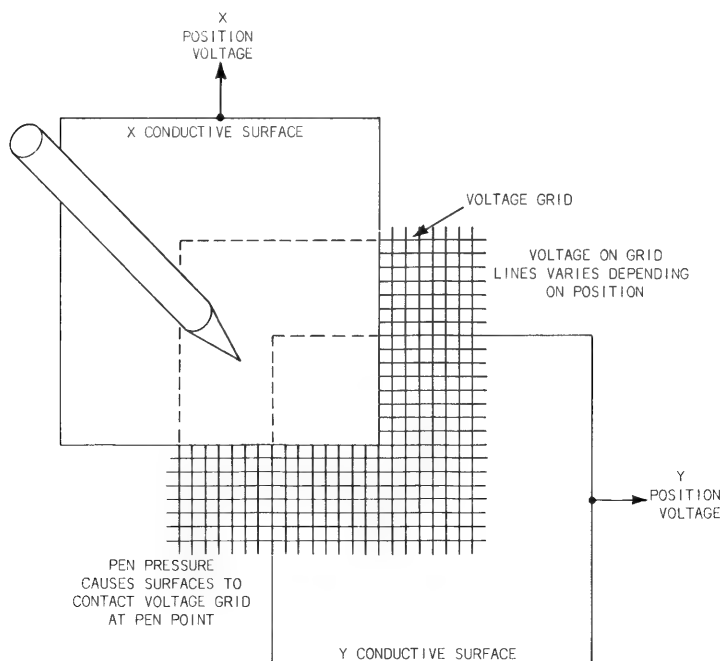   techniques.

Fig. 7-5. X-Y tablet using conductive surfaces
          voltage grid sandwich technique.

then presents the display, less the erased portion, on the next refresh cycle. Since each refresh cycle takes 1/30 of a second, the operator sees the erase as instantaneous.

Often selective erase is part of an erase-write cycle. If an error is noted on the display -- say a word is misspelled -- the cursor is positioned over the incorrect character by the keyboard or graphical input. The operator then presses the correct character. This operation automatically enters the new character in memory, erasing the incorrect character.

Selective erase is a valuable tool for text editing -- e.g. the ability to scan a page of data on the display and make corrections. For most other applications it is an *operator convenience*.

With a bistable storage-tube display, "selective erase" itself is extremely difficult to accomplish. Since the display is stored on the CRT phosphor and the phosphor can be erased only in its entirety, a single character, line or section cannot be deleted without erasing the whole display. However, corrections *can be* made by keyboard or graphical output at the end of a line, on another line or by Xing through the unwanted character. The deletion or change information is sent to the computer and interpreted according to software instructions. The display is corrected the next time it is erased, and rewritten.

For program debugging and problem solving, the ability to have a record of corrections at the end of the corrected line turns out to be an advantage. The display history is a valuable learning tool. The man-machine interaction is enhanced by a display of what has transpired in the conversation. In most applications, corrections can be made, the screen easily erased, and rapidly rewritten.

The job of the time-sharing computer terminal is to bring the computer to the level of the user. The keyboard and terminal graphical output devices are the means for questioning and responding to the computer's offerings. The flexibility of these devices augments man-machine interaction.

DIGITAL-TO-ANALOG AND ANALOG-TO-DIGITAL
CONVERTERS AND VECTOR
AND CHARACTER GENERATORS

The cathode-ray-tube beam is deflected by a voltage
on deflection plates (electrostatic deflection) or a
current through a deflection yoke (magnetic
deflection). The magnitude of this voltage or current
determines the amount of deflection of the beam, hence
the position of the spot on the screen. To address
the many points on the screen, many voltage levels
must be applied to the deflection system. Thus, the
deflection system requires *analog* voltages. A digital
system, with only two voltage levels, would be capable
of selecting only two points on an axis.

The deflection information in the computer is in
digital form. Since the display requires more than
two points on an axis, the digital position
information must be stored in a number of digital
devices. If the devices are connected to form a
register, then 2 raised to the power of the number of
digital devices determines the number of permutations.
For instance, if the digital position information is
in the form of a 10-bit digital word then there are
2^{10} or 1024 possible positions.

In the time-sharing system the digital computer word
is transmitted over the phone line serially and then
converted back to parallel form as described in
Chapter 5. The equipment in the terminal must now
convert the digital information to analog for
deflection of the cathode-ray-tube beam. The
terminal must contain the digital-to-analog converters.
Information at the terminal is often in analog form
and must be converted to digital, before it can be
sent back to the computer. Here, special converters
are required. These are described in Chapter 7.

The vector and character generators are special
digital-to-analog converters that convert the digital
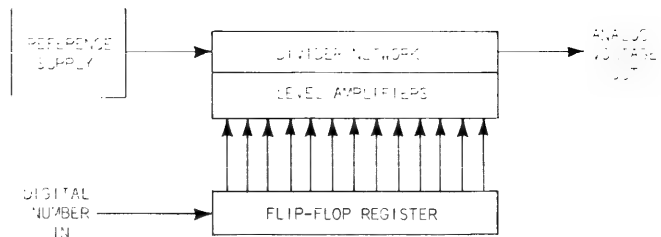computer word to a *sequence of analog deflection
voltages.*

Fig. 8-1. Digital-to-analog converter.

This chapter deals with digital-to-analog and analog-to-digital conversion and the use of such in the remote terminal.

digital-
to-analog
conversion

One method of converting a digital number to an analog voltage uses a resistive divider network connected through parallel lines to a flip-flop register that holds the digital number. See Fig. 8-1. The divider network is weighted so that each bit of the flip-flop register will contribute to the analog output voltage in proportion to its value.

The magnitude of the digital input number determines the *amplitude* of the *analog output voltage,* since the divider network is simply a passive element. However, because digital-voltage levels are not as precise as required in an analog system, level amplifiers are usually placed between the flip-flops and the divider network. The amplifiers switch the divider network between ground and a reference voltage supplied by a precision reference supply.

ladder
adder

The divider network is a ladder adder as shown in Fig. 8-2. The open-circuit output voltage is the sum of 1/2 of any voltage at A, 1/4 of any voltage at B, 1/8 of any voltage at C and so on. Since the voltages at A, B and C are the result of the presence of a bit in the flip-flop register, the output voltage is a properly weighted sum of the input binary bits. The binary bits activate electronic switches to connect the precision supply to points A, B, C, etc.

computer
word

Chapter 5 describes the computer "word" and how the computer uses the word. The monitor buffer was explained, showing how the large computer word could be broken into sections called "bytes."
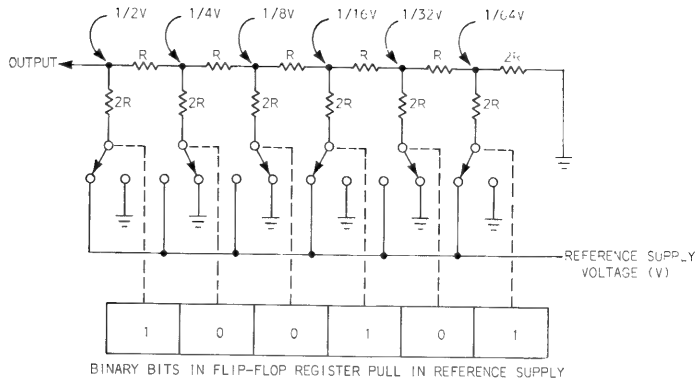
Fig. 8-2. Ladder adder and reference supply.

Most computers work with words longer than 12 bits. For example, a computer might use a 24-bit word. Twelve bits of the computer word are used by the computer to instruct the monitor buffer. The other 12 bits are the "data bits" that are sent to the remote terminal.

In Chapter 8 we are concerned with how the display terminal gets its instruction from the data bits.

In the following discussion, data words being sent to the terminal are of a *12-bit length*. This length is selected because it is compatible with the word length required by the widely used teletype machines. Any display terminal capable of operating with a 12-bit word and the ASCII coding could be operated using the computer software that already exists to talk to teletype terminals.

As was mentioned previously, words are divided into sections. Each section has a specific function. The size of a section can vary from word to word depending on its function. This could cause confusion for a terminal control that must interpret the incoming data word. Therefore, one section of the word is always the same length and in the same position in the word. The bits in this section are the "mode-select code" bits. They allow the terminal to interpret and respond to the rest of the word. Usually, terminal control responds by routing the word to the unit designated by the code, but some functions (space, carriage return, line feed, delete and form feed) are performed by control itself.

(A) CHARACTER GENERATION WORD

(B) X OR Y LOAD WORD FOR POINT PLOT MODE
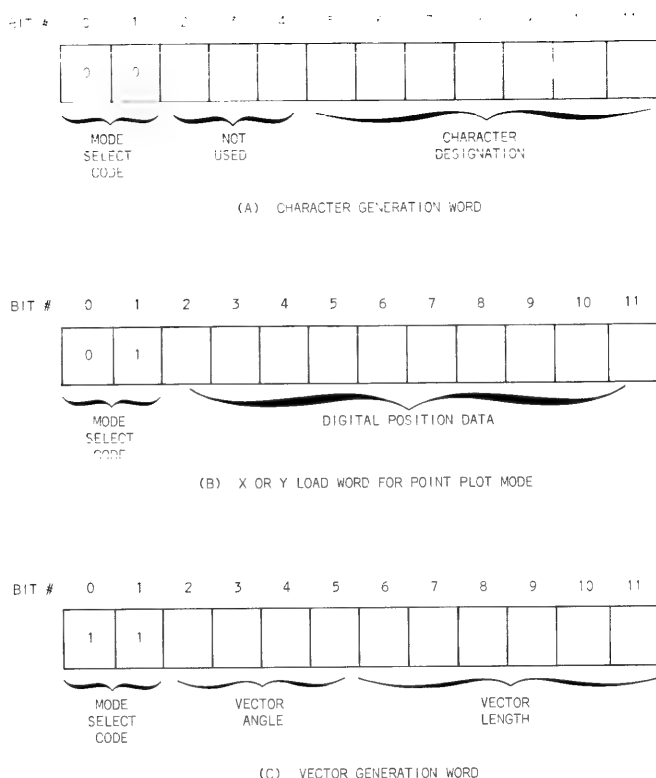
(C) VECTOR GENERATION WORD

Fig. 8-3. Computer words for terminal operation.

In our 12-bit transmitted word, Fig. 8-3, we shall designate the first two bits as the "mode-select code" bits. With two code bits there are four possible word combinations (00_2, 01_2, 10_2, 11_2). "00_2" is the code designated for alphamerics, "01_2" the code for Y LOAD, "10_2" for X LOAD and "11_2" for vector generation. (X and Y LOAD is the terminology for X and Y positioning.) If more than four devices were to be controlled by the terminal, more code bits would be required.

point-
plot
mode

In a "point-plot" mode of operation the computer words are used to plot individual points on the display. The points can be arranged so they form straight lines (often called vectors), curves or even characters. Point plotting is the most flexible plotting mode, because the placement of dots can be completely random.
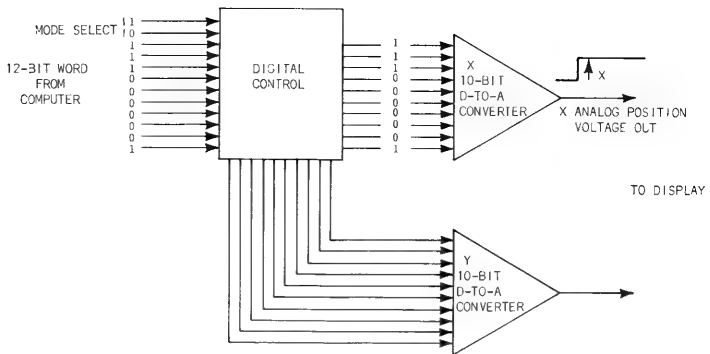
Fig. 8-4. X load in point-plot mode.

There is no limitation on the size of character,
vector or curve; nor is there any restriction on where
each is placed (within the resolution limits of the
display device, and the address limits of the system).

With the 12-bit word we have chosen, the point plot
is accomplished by calling up an X LOAD, then a Y LOAD
and then a Z-axis command (discussed in the next
paragraph). The X-LOAD word consists of the "10_2"
mode-select code, followed by 10 bits of digital
position information. The mode-select code instructs
terminal control to route the 10 bits of information
to the X digital-to-analog converter, where they are
transformed into an analog position voltage (Fig. 8-4).
With 10 bits of digital data, there are 1024 possible
analog level outputs from the digital-to-analog
converter. The analog voltage selected by the word is
output to the display to deflect the cathode-ray-tube
beam in the X axis. The Y LOAD performs the same
function for Y positioning. A combination of X LOAD
and Y LOAD correctly positions the beam.

Two 10-bit words have deflected the beam, but it is
not yet turned on. This can be done in a number of
ways. The computer could also send a Z-axis "ON"
instruction every time it wished to unblank the beam.
The use of a complete word for this function is
uneconomical in terms of transmission and computer
time, and is not usually done. The computer might
send only nine bits of the X or Y LOAD word for
position, with the 10th bit a Z-axis bit. This
method reduces analog resolution, since with nine
bits we have only 512 analog voltage addresses. A

more economical way would be to send an instruction word to the terminal control to tell it to turn on the Z axis after every Y position LOAD. When sending plotting information in this mode, the X position is sent first, then the Y -- the Z axis comes on automatically. This mode has a drawback, in that if we want to move horizontally (X) without a change in the vertical (Y), a Y LOAD must be sent to get the Z axis turned on. The increased resolution (using all 10 bits for position) usually outweighs this limitation.

We have seen how in the "point-plot mode" at least *two* and in some cases *three computer words* are required to plot *one point* on the display. Using this mode to write characters, where an average character was made up of 20 dots, would require at least forty 12-bit words. This time-consuming and costly way of plotting vectors and characters is the reason for the development of vector and character generators.

vector
generator

With the vector generator, only *one* 12-bit word is required to draw one vector. A sacrifice is made with respect to the point-plot mode, in that the vector must now be in one of a fixed number of directions and of a limited number of lengths. The vector generator "knows" how to plot these limited vectors and thus needs only one word to tell it which vector to draw.

A look at the computer word and the vector-generator block diagram will show how this is done (Fig. 8-3 and 8-5).

Mode-select code "11_2" calls up the vector generator. Computer word bits 2 to 5 are fed to a four-bit angle matrix and designate the vector angle. Since four bits do this job, there are 16 (2^4) possible vector angles or each 22-1/2° around the compass. The remaining six bits (6 to 11) determine the vector length. 2^6 or 64 lengths are possible.

The heart of the vector generator is a clock and an X
and Y UP-DOWN counter. The UP-DOWN counters have a
plus and a minus input. If a clock pulse is received
at the plus input, the counter adds one. If a clock
pulse is received at the minus input, it subtracts one.
The X and Y counters output to the X and Y digital-to-
analog converters so that the converter's analog
voltage output reflects the count in the UP-DOWN
counter. Each clock pulse into the counter will then
cause the counter to count and the beam to step.

The rest of the vector generator is necessary to
interpret the 12-bit computer word to draw the proper
vector.

The angle matrix accepts the four-bit angle code and
turns on the proper gates in the gating logic to allow
the clock pulses to go to the UP-DOWN counters. If
the vector were drawn vertically, only the Y counter
would be gated. A 45° angle vector would require
gating of both X and Y counters. A divide-by-two
circuit within the gating logic extends the angle
selection to 16 by gating one of the counters twice
while the other is gated once.

The six-bit "length" counter accepts the rest of the
computer word. Since the counter holds six binary
bits, it can accept any binary number from 0 to 63.
As each clock pulse is gated to the UP-DOWN counters,
it also goes to the length counter, where it subtracts
one count from the number in the counter. When the
number in this counter reaches zero, the proper number
of clock pulses have been gated into the UP-DOWN
counters. The number in the length counter then
determines how many dots long the vector will be. The
clock and logic circuits combine to turn on the beam
after each move. When the length register is counted
to zero, the vector of correct length and angle has
been drawn. *One* computer word has created a vector
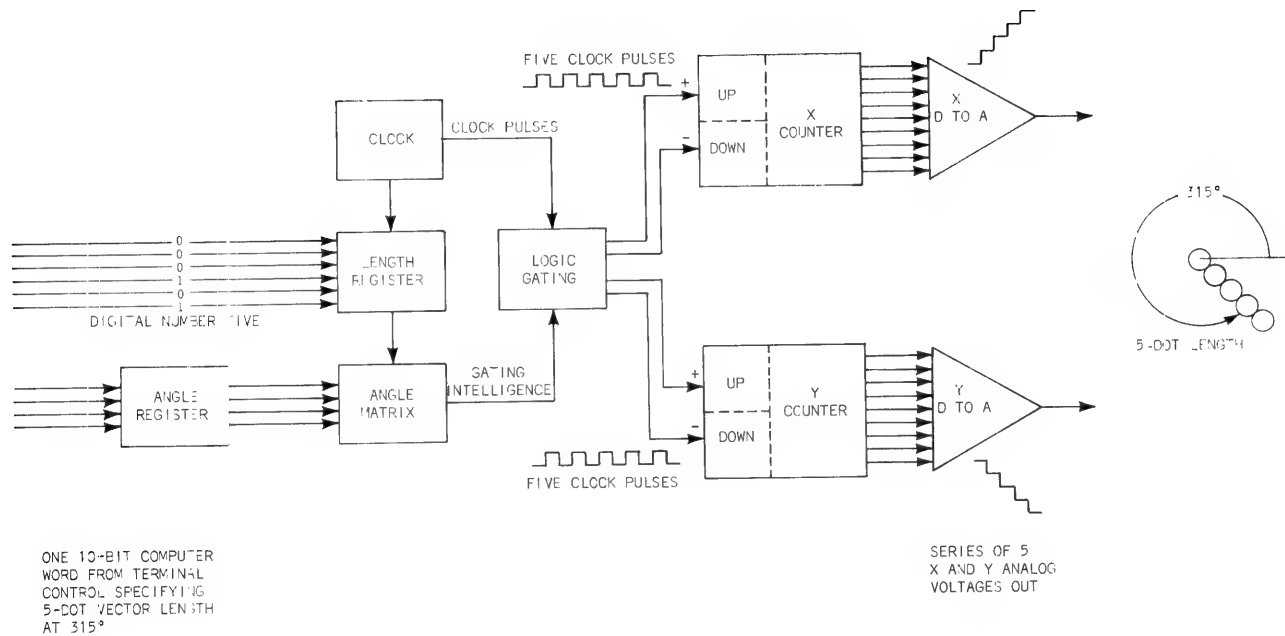that points in any one of 16 directions, and is up to
64 dots long.

FIVE CLOCK PULSES

CLOCK

CLOCK PULSES

LENGTH
REGISTER

0
0
0
1
0
1

DIGITAL NUMBER FIVE

ANGLE
REGISTER

ANGLE
MATRIX

GATING
INTELLIGENCE

LOGIC
GATING

UP

DOWN

X
COUNTER

X
D TO A

UP

DOWN

Y
COUNTER

Y
D TO A

FIVE CLOCK PULSES

315°

5-DOT LENGTH

ONE 10-BIT COMPUTER
WORD FROM TERMINAL
CONTROL SPECIFYING
5-DOT VECTOR LENGTH
AT 315°

SERIES OF 5
X AND Y ANALOG
VOLTAGES OUT

Fig. 8-5. Vector generator.

A vector-plot transmission would be as shown in
Fig. 8-5. Twelve bits are received in serial form
from the phone line. The mode-select code sends 10
bits in parallel to the vector generator. Four bits
go to the angle matrix, six bits to the length
register.

The vector generator creates a series of outputs from
the UP-DOWN counters to the digital-to-analog
converters. Each of the outputs in this series causes
a string of dots to be drawn, making up the vector.

dot-
character
generator

Dot-character generation is somewhat like vector
generation, in that the idea is to use one computer
word to get a complete character, rather than two
words for each dot in the character. (Point-plot mode.)
Instead of the hardware transforming the computer word
to vector angle and vector length, the character-
generator hardware converts the word to dot patterns
that are "known" to the generator. If the CRT beam is
moved to a position, unblanked, moved to another
position, unblanked, moved to another position,
unblanked, etc., dots can be arranged in such a
pattern that the results form a letter, number or
punctuation mark (collectively called characters).
Random movement of the beam between dots for each
character is impractical, since this would be
essentially a point-plot mode and would require
several 12-bit words for each dot. Instead, the beam
is made to step through a fixed pattern, usually a
rectangle of 5 x 7 or 7 x 9 dot positions. The dot
rectangle is called a "dot matrix." The character is
formed by turning on the beam at selected positions as
the beam is stepped through the matrix. The blanked-
dot rectangle is longer in the vertical axis than in
the horizontal axis to give the characters the correct
aspect ratio. For example, nine dots vertical and
seven dots horizontal make up a 7 x 9 dot matrix.

The number of dot positions in the dot matrix
determines the quality of the representation of the
character. If a character is to be written with a
2 x 3 dot matrix, there are only six points that can
be turned on to create a character. Try generating a
pleasant-looking alphabet with this many dots! The
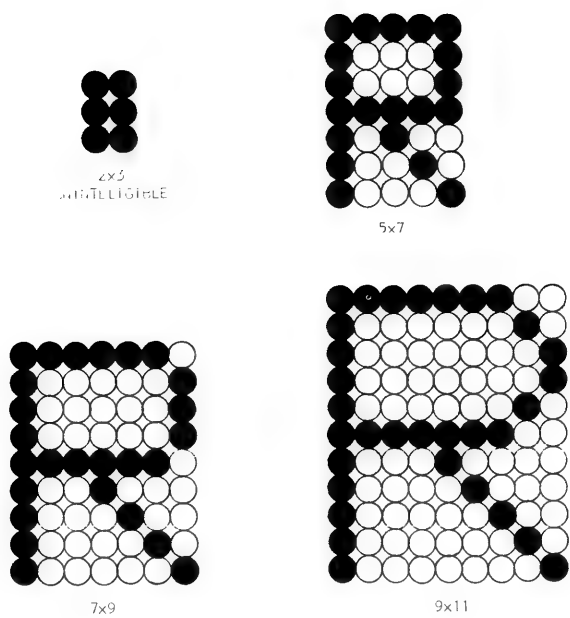greater the quantity of dots used in the matrix, the

Fig. 8-6.   The letter "R" generated from dot
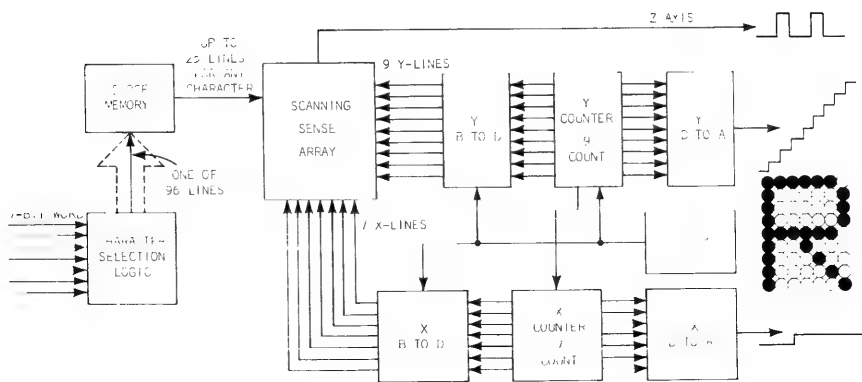matrices.

Fig. 8-7.   Dot character generator.

more pleasing is the appearance of the characters;
but also the more complex the circuitry, and the more
time it takes to step the matrix to draw the character.
For dot counts above a 7 x 9 matrix, the character
quality does not increase enough to warrant the
increased equipment cost and character-writing time
(Fig. 8-6).

The job of the character generator is then one of
stepping the cathode-ray-tube beam through a matrix
of dot positions; and, as the beam is stepped,
turning it on at the correct positions to generate
the character called for by the code from the
computer. Characters can also be generated by
"stroke" rather than dot generators as described
later in this chapter.

character-
generator
word

The character-writing mode of operation is called up
by a "00_2"code in the first two bits of the 12-bit
transmitted computer word. Bits 5 to 11 (seven bits)
are routed by terminal control to the character
generator to designate the character to be written.

The character-generator block diagram is shown in
Fig. 8-7. The generator consists of character-
selection logic, diode-memory matrix, 7 x 9-dot
scanning sense array, digital clock, nine-step Y
counter, seven-step X counter, X and Y binary-to-
decimal converters, and X and Y digital-to-analog
converters.

Character-generator operation is best described in
three parts: First, how the blanked beam is stepped
through the 63 positions making up the character
rectangle; second, how the seven-bit character code
is interpreted and used to turn on the beam at the
correct positions to write the character; and, third,
how the beam is moved over in position to start the
next character without additional computer words.

beam
stepping

Beam stepping is done by the digital clock, nine-step
Y counter, seven-step X counter and the X and Y
digital-to-analog converters.

The digital clock is an oscillator whose output is
squared up to provide a clock pulse for each
oscillation. The output of the clock is sent to the
nine-count Y counter and the X and Y binary-to-decimal
converters.

The Y counter counts from one to nine and then resets. Each clock-pulse input causes the counter to increment.

The X counter counts from one to seven and then resets. Each reset pulse of the Y counter causes the X counter to increment. The X counter then increments each time the Y counter has counted up to nine.

The Y counter outputs to the Y digital-to-analog converter. Likewise, the X counter outputs to the X digital-to-analog converter.

The Y digital-to-analog converter changes each number in the counter to a vertical analog-position voltage. As the counter counts, analog-voltage steps are applied to the vertical deflection plates of the terminal display.

The X digital-to-analog converter changes each number in the X counter to a horizontal analog-position voltage. As the X counter counts, analog-voltage steps are applied to the horizontal deflection plates.

The sequence is one of the clock driving the Y counter for nine counts, at that time the Y resets. The reset increments the X counter one count. Then there are nine more Y counts, the X increments and-so-on. If the beam were unblanked, the display would appear as a vertical row of dots, a horizontal shift and then another vertical row of dots, a horizontal shift, etc. The counter outputs thus cause the X and Y digital-to-analog converters to output analog voltages that step the beam in the pattern of the 7 x 9-character rectangle. The first requirement for character generation has been met. The blanked beam has been stepped through 63 positions.

character selection

The other output from each of the X and Y counters is to the X and Y binary-to-decimal converters. The Y binary-to-decimal converter has nine output lines, the X binary-to-decimal converter seven. As the Y counter counts from one to nine, the Y binary-to-decimal converter will have outputs first on line one, then two, then three and so on up to nine. The X binary-to-decimal converter acts in the same manner but has only seven lines. These output lines are connected to 7 x 9 *scanning-sense array* to provide information as to the beam position on the display. This is necessary to assure that the proper dots can be

unblanked to draw the character. Before we get into
this discussion, let's see how the other inputs to the
scanning-sense array originate.

Recall that the character selection is made by a
seven-bit code from the computer. Each combination
of those seven bits will cause a unique character to
be written.

The seven-bit word is entered into a register within
the character-selection logic block. The code in the
register goes to the symbol-select matrix, also within
the logic block. The symbol-select matrix interprets
the seven-bit code and selects the proper character
line. There are 96 possible character lines, since
there are 96 characters in the ASCII code. The seven-
bit code is capable of 128 unique selections; the
difference $129 - 96 = 32$, are used for special commands,
instead of characters.

diode
memory

The character lines are connected to a diode memory.
When a line is activated, a number of diodes connected
to that line are put on conduction. Each of these
diodes connects to the 7 x 9 scanning-sense array. It
takes from 4 to 25 diodes to make a character,
depending on its complexity. The average character
takes 16.7 diodes. Since there are 96 possible
characters, and an average of 16.7 diodes per
character, the diode memory contains about 1600 diodes.

This group of diodes is called "read-only memory"
because, whenever one line of the 96 is activated,
specific diodes transfer the information to the
scanning-sense array. The diode circuitry has
"memorized" -- it is wired to connect specified diodes
when a character line is activated.

scanning-
sense
array

The scanning-sense array is made up of 63 triple AND
gates, one gate for each dot position in the character
matrix. When the inputs from the X and Y binary-to-
decimal converters (indicating dot position) coincide
with the character information from the diode memory,
the scanning-sense amplifier outputs a pulse to turn
on the Z axis of the display.

An information-in information-out sequence would start
with the arrival of a seven-bit word at the input
register. This seven-bit word would cause the
generator to unblank designated dot positions as it

stepped through its 7 x 9 matrix. The result is a character created from up to 25 of those 63 dots. One computer word then creates a pattern of up to 25 dots that form the character.

stroke-character generator

The operation of the stroke generator is similar to the dot generator, in that it must output X and Y analog deflection voltages and turn on the Z axis at the correct time. However, instead of holding the beam steady as it is unblanked, the stroke generator moves it to create a line or "stroke."

The data word is decoded in symbol-selection logic circuitry, much as in the dot-character generator. From there on the operation is different. The character-select line must choose which "strokes" will be required to create the character. The stroke is drawn by one operational amplifier driving the X and another driving the Y of the display unit. Two operational amplifiers are required for each different stroke (except for strokes that are purely vertical or purely horizontal that require one operational amplifier). Timing of the strokes is controlled within the generator. The more possible strokes, the better looking the character (just as the more dots the better looking the dot character). Generally, dot generators can present a better looking character for a given complexity. However, stroke generators tend to permit faster operation; thus in refreshed systems, stroke writing is usually used.

dynamic intensity

Normally, in display applications, a constant deflection and writing rate can be maintained. In some graphical applications and often in *stroke-character* generation, variable deflection speeds are necessary. These varying speeds require a change in beam intensity to prevent the stored image from having a thicker stored trace at the slow-moving sections. These beam changes are usually controlled by stroke generators.

Controlling the cathode-ray-tube beam current via the grid, requires a very stable driving circuit and imposes a basic limitation of changing spot size as a function of beam current. It is more economical to design switching circuits as opposed to wide-range stable analog-voltage control circuits. For these reasons, it is desirable to control the duty cycle of the beam rather than the beam current to maintain a

fixed charge per unit area. The stroke-generator
circuitry must determine the deflection velocity of
the beam and use it to control the number of beam-on
pulses per second, thus keeping the charge per unit
area scanned constant regardless of beam velocity.
This makes stroke-character generation considerably
more complex than dot generation for use with storage
displays.

Dot-character generation is more compatible with
storage display units than stroke generators for
several reasons. The cost is usually less since the
operation is essentially digital until the output
where it is converted to analog. Stroke generators
must contain more complicated analog circuits to form
the strokes. The width of a stored trace depends,
among other things, on the time the beam is left
unblanked at a given point. As mentioned above, this
creates problems for the stroke generator. The dot-
character generator lends itself nicely to setting up
the unblanking for each dot in the matrix to correspond
to the dot-writing time of the display unit. (For an
explanation of Dot-Writing Time, see Chapter 9.) For
example, the Tektronix Type 611 Storage Display Unit
has a dot-writing time of 20 μs. With a 7 x 9 matrix
the generator would step through all 63 positions in
1.26 ms (20 μs x 63-dot positions) = 1.26 ms. This is
the maximum time it would take to write a character.
Since most characters are drawn by unblanking only 5
to 25 of the 63 dots, a considerable speed-up in
character-writing time can be had by stepping the
nonwritten dots in a minimum time, say 5 μs, and
pausing the full 20 μs on the dots to be written.
This, of course, requires additional logic in the dot
generator, but the saving in text-writing time is
usually worth the expense.

The terminal user often wishes to perform an operation
on the data on display. He might want to delete or
add part of a graph or line. To alter a point on the
display, the computer must know the X and Y
coordinates of the point. These coordinates are best
represented by X and Y analog voltages. To transmit
the value of those voltages to the computer, the
analog voltage must be converted to digital form.
analog-   This is done in the analog-to-digital converter. Most
to-digital   terminal-to-computer devices would then contain a
conversion   mechanical-to-digital or analog-to-digital converter.

The basic unit of analog-to-digital conversion is the comparator circuit. This circuit compares an unknown analog voltage with a reference voltage, and indicates which of the two is larger.

There are a number of methods employed for analog-to-digital conversion; simultaneous, feedback, counter, continuous and successive approximation. Only the most basic types will be discussed here.

simultaneous method

The simultaneous converter is representative of how analog-to-digital conversion is performed and is easiest to understand. Fig. 8-8 shows a simultaneous converter using several comparator circuits. Each comparator has a reference input signal. The other input terminals of the comparators are driven by the unknown analog voltage we wish to convert. The comparator will turn "on" if the analog input is larger than the reference input. Then, if none of the comparators are on, the analog input must be less than $V/4$. If comparator 1 is on, comparator 2 and comparator 3 off, the input is between $V/4$ and $V/2$. Similarly if comparator 1 and comparator 2 are on, comparator 3 off, the voltage is between $V/2$ and $3V/4$; and if all comparators are on, the voltage is greater than $3V/4$.
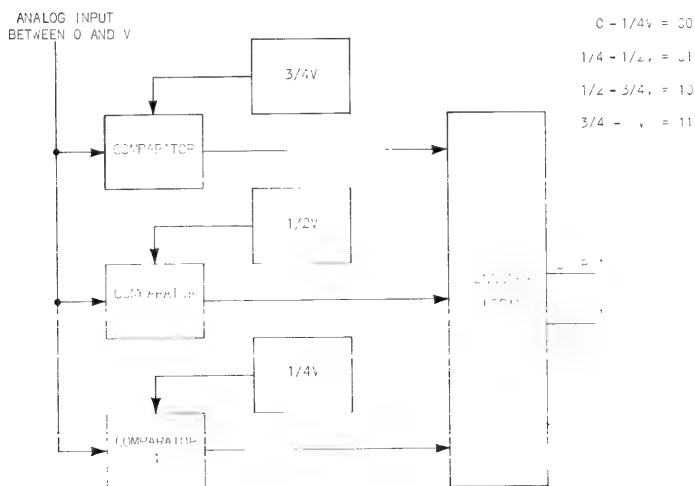
Fig. 8-8. Simultaneous analog to digital converter.

With this analog-to-digital converter, the voltage range V is divided in four parts, which are coded in the coder to give two binary bits of information 00_2, 01_2, 10_2, 11_2. Seven comparators would give three bits of binary information. Fifteen comparators would give four bits. In general 2^n-1 comparators will give n-bits of binary information.

resolution

Obviously the number of discrete analog levels that the comparators can detect determines how accurately the digital information is represented. In other words, how closely the analog voltage can be "resolved."

The simultaneous method is extremely fast for small-resolution systems. For larger-resolution systems (more bits), this method requires so many comparators that it becomes unwieldy and uneconomical.

feedback
converter

If the reference voltages were variable, only one comparator would be needed. Each of the possible reference voltages could be applied in turn to determine when the reference and analog input were equal. But a digitally controlled analog reference is simply a digital-to-analog converter. Thus the feedback converter of Fig. 8-9 is actually a close-loop feedback system. The main components are the same as a digital-to-analog converter. The comparator indicates whether the corresponding analog voltage is larger or smaller than the unknown analog input voltage. With this information, the digital number is modified and compared again.
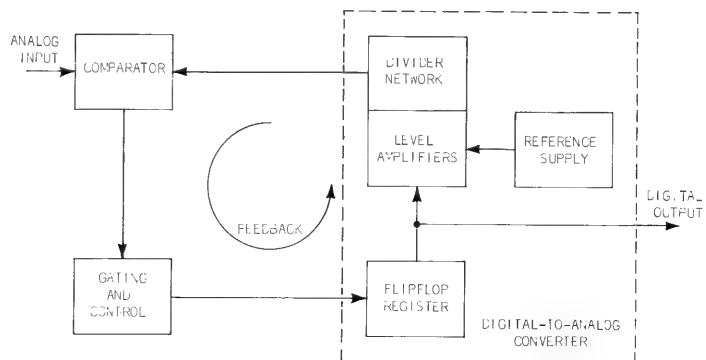
Fig. 8-9. Feedback analog-to-digital converter using digital-to-analog converter.

Numerous ways may be used for controlling the
conversion. The simplest is to start at zero and
count until the digital-to-analog converter output
equals or exceeds the analog input.

The counter, continuous and successive approximation
methods of analog-to-digital conversion are variations
to the feedback method. Each method uses different
gating and control circuits and flip-flop registers to
optimize for specific applications.

The reason for the existence of so many types is the
differences in price for a given speed and resolution.
Obviously the sequential-feedback type is slower than
the simultaneous type, but is cheaper than the
simultaneous type when high resolution is needed.

9

CHARACTERISTICS AND SPECIFICATIONS
OF DIRECT-VIEW BISTABLE STORAGE TUBES

Resolution, brightness, contrast, writing speed,
information display rate and CRT life are of primary
interest to the user of the direct—view bistable
storage tube as an information display device. Since
these characteristics describe the performance of the
device, the limits of this performance are specified.
This chapter describes storage-tube characteristics
and the specification of those characteristics.

resolution

In the field of optics, "to resolve" means to be able
to make visible the individual parts of an image.
Once the image is broken into the maximum number of
visible parts, those parts can be counted. The count
becomes the "resolution" of the image. Resolution is
then defined as the degree to which separate but
adjacent elements of an image can be distinguished.

The term "resolution" is usually abstracted further
when we talk about the *resolution* of a piece of
equipment.

If the object that the image is created from has an
infinite number of parts, it is only the apparatus we
are viewing or presenting it with that limits the
number of individual parts we can make visible. The
term "resolution" then gets tagged onto the equipment.
We mean "the equipment's ability to resolve."

This characteristic of any display unit, the "ability
to resolve" or "resolution" is very important to the
user. He is usually not interested in the *resolution
number* per se, but in the fact that it tells him what
an image will look like when presented on the display.
Will it look to the eye exactly like the original
object (example -- good quality film); or will it be
grainy, be made of visible lines or dots; or will it
be only an outline, a suggestion of the object.
Another consideration is whether symbols such as
letters of a given size on the display will be legible.
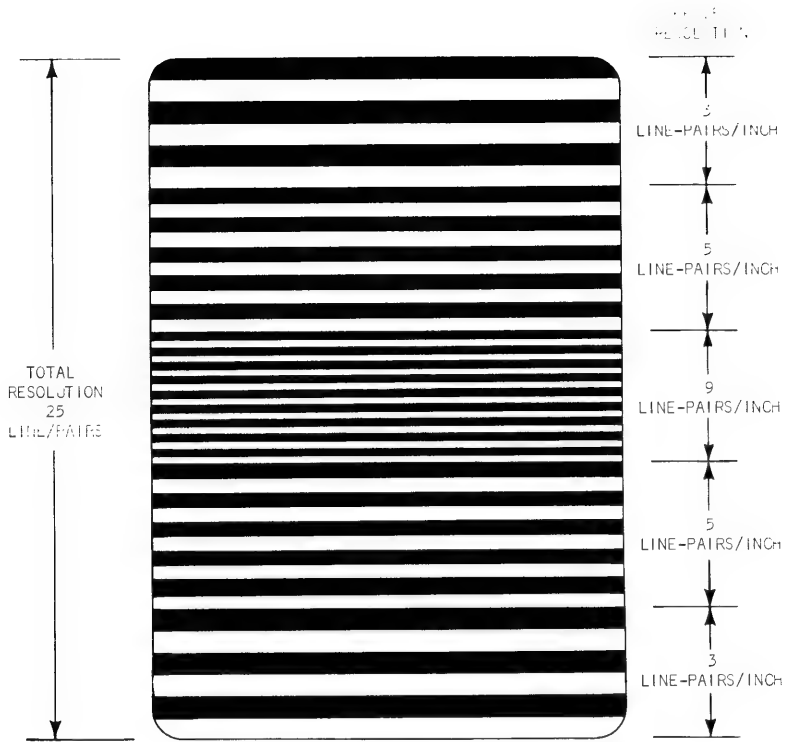
Fig. 9-1. Specific resolution and total resolution expressed in line-pairs. (Could also be expressed in lines or in lengths other than inches -- cm, mm, etc.)
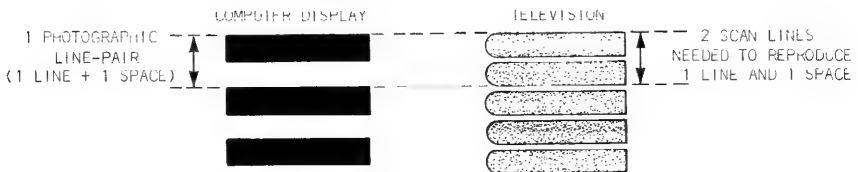

Fig. 9-2. Lines versus line-pairs.

It is important, then, that the specification describing the resolution of a display unit convey to the user what the images on the display will look like. As might well be imagined, this is a difficult task. Perhaps this was the point where the phrase "a picture is worth a thousand words" originated.

specifying
resolution

A natural result of this difficulty is that a multitude of methods of specifying resolution has evolved. Almost every discipline measures and specifies resolution differently. The evolution is such that the method used in each discipline conveys to the *educated* user what the display will look like. The differences can usually be described to allow comparisons to be made.

To specify resolution a definition of terms is necessary. Is the resolution in lines, line-pairs, lines per inch, lines per mm, line-pairs per inch, etc.? What do these terms mean, what is the difference between them?

Resolution specification in units per unit-distance along the vertical and horizontal axis such as lines per mm, is known as "SPECIFIC" resolution. Resolution specified as a number of units for a full display is called "TOTAL" resolution (Fig. 9-1). In cases where the resolution is uniform for the whole display, the specific resolution (units per unit-distance) times the number of distance units will yield the total resolution. In computer displays the specific resolution used to calculate the total resolution should be the specific resolution in the corners, since the corners must be as easily resolved as that in the center. (As opposed to television where the corners are allowed to be poorer than the center.) The units of resolution are usually "lines" or "line-pairs" (Fig. 9-2). The term "line" is used where scanning devices are employed (TV and facsimile). It takes two scan "lines" to produce one line and one space.

"Line-pair" is defined as a line with an adjacent line space, the width of the line being equal to the line space. "Line" and "line-pair" are often used interchangeably, resulting in considerable confusion.

The standard photographic unit, for instance, is "lines per millimeter" but actually means lines *and spaces* per millimeter and should therefore be referred to as *line-pairs* per mm.

The bistable storage-tube display, when hooked to a computer, is not normally used in a raster mode, so Tektronix has chosen to specify resolution in line pairs to avoid confusion with raster-type displays. Tektronix also specified *total* resolution, but it should be understood that this figure is derived from a specific resolution that is minimum for the full screen.

vertical-
versus-
horizontal
resolution

There is no question but that square resolution, i.e. equivalent specific resolution in the vertical and horizontal directions, is desirable in the reproduction of graphics; due to the makeup of the human eye.

Experiments have indicated that the human eye sees slightly less resolution vertically than it can horizontally. However, the degree is so slight the phenomenon may be dismissed as insignificant.

Remember, the resolution discussed here is *specific* resolution (line-pairs per unit-distance). If the aspect ratio of the display device was such that it was longer in one axis, then the *total* resolution in that axis would of necessity be larger. For example, the Tektronix Type 611 Storage Display Unit has a long axis of 22 cm, and a short axis of 16 cm; an aspect ratio of approximately 4/3. The *total* resolution of the two axis is 400 x 300 line pairs, essentially *square specific resolution.*

picture
elements

It is often convenient to express resolution in "total number of picture elements." This can be determined by multiplying the horizontal resolution times the vertical resolution. When a "line-pair" resolution is used we must recall that there is actually a line (called a line space) between each written line that is also a picture element. We must, therefore, double the number of line pairs on an axis to get the total number of elements. The Tektronix Type 611 Storage Display Unit would then have 800 elements vertically, and 600 horizontally for a total of 480,000 picture elements. This compares with other media as follows:

IMAGE	RESOLUTION (Total Elements)
3-3/4 x 5-inch newspaper halftone (about 50 dots per inch)	50,000
8-mm home-movie frame	50,000
525-line commercial TV	150,000
1000-line closed-circuit television	550,000
8 x 10-inch glossy contact print from fine-grain negative	150,000,000

readability criteria

What are the criteria for determining whether an alphameric display has adequate resolution?

In the computer display field a standard does not seem to have been established. The microfilm industry is somewhat more standardized and we can glean some facts from their standards and readability studies.

All readability criteria are subjective and/or arbitrary to a degree.

The National Bureau of Standards has a complex formula that deals with microfilm reduction of engineering drawings, and a quality factor; all of which is not relevant here. However, we can relate to the conclusion they draw using the lower case "e" of a height of 2.4 mm (3/32 inch). Excellent quality (legibility) should be obtained if the character is made up of 8 line-pairs. One of marginal quality can be obtained using 5 line-pairs.

Studies conducted by the Battelle Memorial Institute of Columbus, Ohio on the readability of material for long periods, and for *pleasure*, indicate that characters of the size just described (3/32" lower case "e") should be a minimum of 7 line-pairs and are optimum at greater than 17 line-pairs.

Other studies at Bell Labs and Eastman Kodak indicate that 8 to 10 line pairs is about the norm for existing microfilm-enlarged characters.

TV studies constructed around having subjects judge
the readability of printed characters in TV images at
various scan-line definitions seem to establish 10 as
the minimum number of scan lines required for *assured
legibility* of a given symbol. This would be
equivalent to about 5 *line-pairs* in a computer display.
The bistable computer display would require fewer line
pairs for the same end result because of its lack of
flicker and display jitter.

Using the line-pair resolution of the Tektronix Type
611 Storage Display Unit shows that the specific
resolution is 1.9 line-pairs per mm which would be
about 6 line-pairs for a character of 1/8" height.
This would indicate that to assure legibility at all
points on the screen, including edges and corners,
characters much less than 1/8" in height and width
should not be used. Experience shows that the Type
611 can write legible characters much smaller than
1/8". This apparent discrepancy is explained later
in this chapter where the Type 611 specification is
discussed.

computer-
display
resolution

In a computer-driven display system, the resolution
of the system cannot be any better than any one of
its parts. Therefore, we must consider the resolution
of the display drivers as well as the display itself.

address-
ibility

The *electronics* of the display system has a resolution
limit of its own, called "addressibility." If 10 bits
of position information for an axis is being
transmitted, then only 2^{10}, or 1024 locations,
"addresses," on that axis are obtainable, regardless
of how good or bad the display unit resolution is.

The addressibility of the electronics is then
determined by the number of bits in the computer word
available for positioning and the stability of the
digital-to-analog converter. If the converter is
unstable, addressibility will be impaired due to the
beam being improperly positioned or moved during
unblanking (Fig. 9-3).
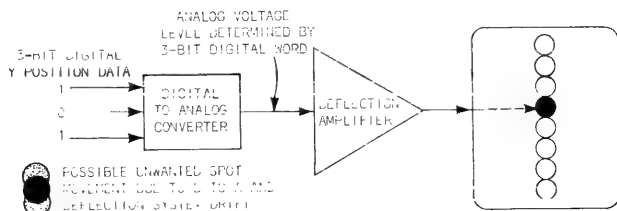
Fig. 9-3. Driving system addressibility.

DOTS WRITTEN AT
COMPUTER LIMIT
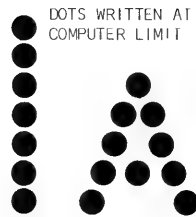
Fig. 9-4.   CRT resolution exceeds driving
addressibility.

Compatibility of display resolution and addressibility
of the driving electronics is necessary. If the
display has poor resolution, it's senseless to have
orders-of-magnitude higher addressibility in the
electronics, as the dots will only be blotched
together. If the display has high resolution, but
the electronics' addressibility is limited, another
problem is created. When fine dots are written in
patterns to form characters, the low addressibility
will space the dots in such a manner that the
characters will be made of dots instead of lines.
Character quality is impaired (Fig. 9-4).

Since high resolution is usually more difficult to
obtain in the display device, this latter case is not
as much of a consideration as the former, except when
trying to make large-size alphamerics with a limited
number of dots from a character generator.

As a general rule, the addressibility of the
electronics should be better than that of the display,
so that dots can be written to overlap one another.
This allows lines to be written in a manner that makes
them look like lines and not a string of dots (Fig.
9-5).

DOTS WRITTEN
AT DISPLAY
RESOLUTION

DOTS WRITTEN AT
GREATER THAN
DISPLAY
RESOLUTION
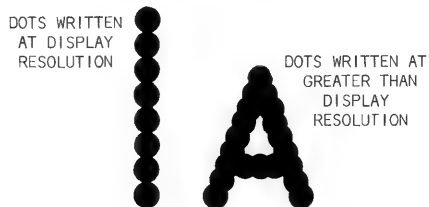
Fig. 9-5.   Driver addressibility exceeds display
resolution.

bistable-
storage-
tube
resolution

A basic factor affecting resolution of any CRT device is the beam spot size. Obviously, the smaller it is, the greater the possible resolution. Changes in spot size affect resolution equally in the vertical as well as horizontal directions *if* the spot is round.

Some storage tubes such as that in the Tektronix Type 564 can be considered essentially grainless, i.e. the spot size is bigger than the phosphor agglomerates. Therefore, the spot size is the primary resolution determinant. In the case of other tubes this is not always so. The Tektronix Type 601 and Type 611 Storage Display Units have a "raised-collector" target structure with about a 6-mil pattern that tends to quantize the stored spot. The advantages of the raised collector far outweigh the quantizing effect it has on the display. Raised collector targets are discussed later in the chapter. Thus the Type 611's minimum spot size is a 6 x 6-mil square, even though the center spot size may be smaller. The Type 601 normally has a spot size larger than 6 mils when writing at the specification writing rate, but at much lower beam currents, near the center of the display, it may be target limited.

In any large-screen tube, the spot size will be larger at the edges and corners in spite of focus-correction circuits. The specific resolution at these points will be worse than that in the center. Since information in the corners is as valuable as that in the center, the total resolution should be calculated taking into account the varying specific resolution across the tube.

It is a characteristic of the bistable storage tube that small particles can become "spontaneously" written (particles fade-up). Also small particles occasionally tend to become unwritten (particles drop-out). If the tube is to keep information stored for some time, then the resolution may be degraded due to fade-up and drop-out.

fade-up

drop-out

Fade-up has a higher tendency to occur when an unwritten area is very small, such as the space between the dot and the stroke of the letter "i". Drop-out is more likely to occur when the written areas are small, such as an isolated period. (For clear legibility "periods" and the "dot" on an "i" should be formed from groups of four matrix dots.

<div style="margin-left:2em">

display
noise

Particle fade-up and drop-out are often classified as display "noise." "Noise" is defined as "anything which is not in the message."

With the bistable storage tube there are four principal noise sources.

(1) Random noise on a recorded trace width due to the phosphor agglomerate variations.

(2) Spots on the CRT which remain written even after erasure. Since most messages use less than 10% of the CRT area, the probability is high a permanently written spot won't coincide with a desired written spot.

(3) Spots on the CRT which remain unwritten after the spot was excited properly with the writing gun (drop-out).

(4) Spots which appear after the message has been written for a period of time (fade-up).

ideal
resolution
specification

It's difficult to say what the ideal specification for every possible use of a computer-driven display might be, but for a majority of cases we can generalize. A specification that would state the maximum number of alphameric characters that would be clearly legible on the full screen should satisfy the alphameric user. The specification would have to spell out the type font (size and style) and the character spacing. Most *graphics* users would also require a specification that stated the maximum number of line-pairs or dot-pairs that could be placed in the horizontal and vertical direction and still be discernable as individual line-pairs or dot-pairs (dot and dot space).

This specification would have to take into consideration whether the target was essentially grainless (continuous with spot size bigger than phosphor agglomerates) or quantized, such as an array of gas discharge cells, that can light up only at discrete points. With the continuous target, dots or lines can be written closer together than the line-pair resolution with the result that the lines or dots touch. In the quantized system this cannot be done.

</div>

We have seen that the computer generates alphamerics
through dots or strokes. For the dot-character display,
the type font might best be expressed in terms of the
size of the dot matrix (50 x 70 mils or 70 x 90 mils,
equivalent to a 5 x 7 or 7 x 9 matrix with a 10-mil
spot size).

actual
resolution

The resolution specification for a given computer-
driven display often varies from the ideal because of
difficulty in measuring the resolution (at high
resolutions the complexity of measuring resolution
often dictates how it is specified) and because
abberrations and target idiosyncrasies require
additional words for clarification.

Type 611
specification

Tektronix specifies the stored resolution of its Type
611 Storage Display Unit as "4000 characters, clearly
legible, with good spacing, based upon a 70 x 90-mil
dot-character matrix. Equivalent to 400 vertical by
300 horizontal stored line-pairs. (Resolution is
measured using 400 x 300 stored dots since closely
spaced line-pairs exceed 25%-incremental storage.)"
This resolution specification is intended to convey
both the *alphameric* and *graphic* capabilities at the
rated writing speed and after 2000 hours of use.
Somewhat better results can be obtained at lower
writing speeds, particularly with a new cathode-ray
tube.

alphameric

The 4000-character specification is intended to convey
the alphameric capability of the display. The
Tektronix Type 611 Storage Display Unit has a screen
size of 8.66-inches high by 6.30-inches wide. The
character size is designated as 90-mils high by 70-
mils wide. We must include space between characters
and lines. If we allow 30 mils between characters and
45 mils between lines each character will occupy a
space 135 x 100 mils. This says we can have 63
characters per line (6.3 inches ÷ 100 mils) and 64
lines (8.6 inches ÷ 135 mils). The total number of
characters is 63 x 64 = 4000.

The 90-mil character height is approximately 3/32".
The readibility studies discussed previously comparing
line-pair resolution to character legibility indicate
that characters much less the 1/8" in height will not
be clearly legible on all parts of the screen.
Actually, the clarity of characters when *first written*
on the Type 611 shows that symbols as small as 1/16"
are clearly visible.

Some computer users will want to view the display for several minutes. To ensure this capability, Tektronix storage display resolution specifications are based on the resolution existing 15 minutes after the display was stored. Over 15 minutes some particles fade-up and others drop-out, so that a character 1/16" in height, though legible when first written might not remain legible on all portions of the screen. (Due to edge defocus, character legibility would be worse in the corners.) The 90-mil (3/32") character size in the specification says that after 15 minutes, these characters are still legible in the corners of the tube.

graphics "The equivalent to 300 x 400 line-pairs" specification is intended to convey the graphic capability. The "equivalent to" portion of the wording is necessary because of a 25%-incremental area restriction. An incremental area of the storage target is a 10 x 10 square, made up of 10-mil dots (100 mils by 100 mils). See Fig. 9-6. If more than 25% of an incremental area is stored (written), the target could have miscollimation. This is not as serious as it might sound because 25% of an incremental area is a very high density. A typewritten page, by comparison, is only 15% incrementally written. It does restrict the graphics user, in that parallel lines at the resolution limit cannot be written. They must be far enough apart to honor the less than 25% incrementally written rule.
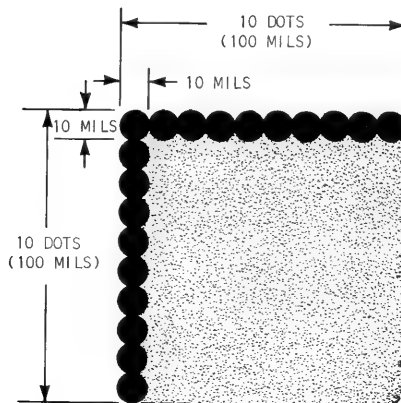
Fig. 9-6. Bistable storage tube -- definition of an "incremental area".

In attempting to measure the line-pair resolution of
the tube, the 25% density is exceeded. If 300
vertical line-pairs were to be written horizontally
across the 16 cm of the tube, each *line-pair* would
have to be less than 21 mils. By definition, each
line would have to be less than 10-1/2 mils. A look
at an incremental area shows that the 100 x 100-mil
area would have every other 10-mil distance written,
and would therefore be *50% incrementally stored*.
Miscollimation would result, causing some target
areas to receive more flood current than others.
This would result in uneven aging called "differential
aging." Also, the probability of fade-up is increased
about a factor of 2, since more of the unwritten dots
will be near written dots. To test the resolution,
the screen can be written with 300 x 400 *dots*. This
verifies that the fineness of dot can be obtained
without exceeding the incremental area restriction.

As one might expect from the readibility discussion,
the line-pair resolution would be equivalent to *more
than* 300 x 400 when first written. After 15 minutes
of storage, the loss of resolution due to fade-up and
drop-out lowers the resolution to the specification.
Some confusion may arise when we say "the resolution
changes over 15 minutes." The spot size of the beam
does not change, new information can be written with
as much resolution as the old. But information that
has been *stored* for 15 minutes may slightly lose
definition. We call this a loss of resolution.

raised
collector
target

The raised collector target used in the Tektronix Type
601 and 611 Storage Display Units tends to quantize the
target as described earlier. This is not a major
effect because the CRT spot size is larger than the
raised collectors. The raised collector target is
included in display units because it aids in
collimation and virtually eliminates "trace shadowing."
Trace shadowing is a phenomenon wherein the edges of
written areas (traces) tend to darken due to the
convergence of flood electrons on the written areas.
In an information display, darkened trace edges degrade
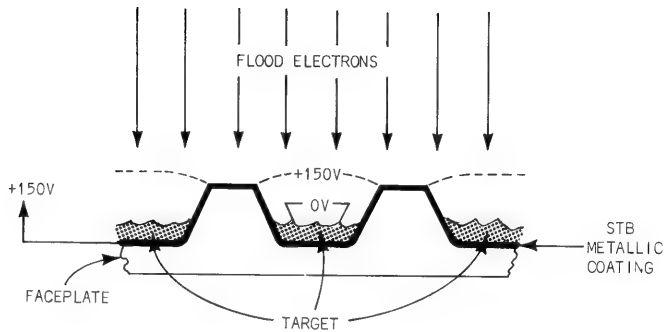
Fig. 9-7. Raised collector target.

the legibility. The raised collector target, as shown
in Fig. 9-7, creates an equipotential surface across
the collector tips. This surface is a very short
distance from the rear of the target surface and
presents a uniform field to the arriving flood
electrons. The electrons then tend to strike the
target perpendicularly; eliminating the convergence
that creates the shadowing.

In conclusion, Tektronix has attempted by specification
to convey to the user both the alphameric and graphic
resolution capability of its display units.

television
resolution

Since computer display "line-pair" resolution is often
confused with television "line" resolution, a
discussion of TV resolution is in order.

In television, resolution is specified as the number of lines in the transmitted raster (Fig. 9-8). This differs from the actual screen resolution that we use in computer display. First of all, the television system uses a portion of the transmitted lines for retrace (raster flyback). A 525-line TV system uses about 40 lines for retrace, allowing only 485 available for viewing.

Resolution in the direction perpendicular to the scanning (vertical in the conventional TV system) is further affected adversely by a number of factors that are conveniently lumped together in what is known as the "Kell factor."

The Kell factor is an expression of the degree of resolution degradation attributed to segmenting of an image by scanning. Its value is arrived at subjectively and has been expressed anywhere from 0.60 to 0.90. However, in recent years, 0.70 has been more or less offically adopted as the value for the Kell factor.

The degradation of 30% that the Kell factor represents takes into account loss of image definition due to spacing between lines, the angle at which a scan-line intercepts a line of image detail and the degree of registration that a scan line overlaps a coincident line of image detail. See Fig. 9-9.

The Kell factor is applicable to resolution only in the direction perpendicular to the direction of scanning. This is because the video signal can appear anywhere along a horizontal line. There is no restriction on video time position along a line -- only in registration of the lines themselves.

The effective on-screen resolution of *525-line* TV system is degraded to *338 lines* vertically by retrace losses and the Kell factor. (525-40 x 0.70 = 338.) The horizontal resolution is about 432 lines. (This is determined by gating a 4-MHz sinewave with a half cycle of 0.125 μs into the 54 μs of unblanking time for a TV line.) The total number of picture elements is about 150,000 (432 x 338).

(A) TRADITIONAL METHODS OF SPECIFYING SCAN-LINE DEFINITION IN FAX AND TV. IN FAX, THE QUANTITY OF LINES CONSTITUTING THE IMAGE IS GIVEN ON A PER-INCH BASIS. IN TV, IT IS GIVEN IN TERMS OF LINES PER FRAME, AND THE COMPLETE SCANNING OF AN IMAGE IS SAID TO CONSTITUTE A "RASTER".

HORIZONTAL RESOLUTION: ELEMENTS RESOLVABLE IN A HORIZONTAL DIRECTION (AFFECTED BY CHANNEL BANDWIDTH WHEN SCANNING IS HORIZONTAL)

VERTICAL RESOLUTION: ELEMENTS RESOLVABLE IN VERTICAL DIRECTION (AFFECTED BY SEGMENTING OF IMAGE WHEN SCANNING IS HORIZONTAL)

(B) SEGMENTING OF AN IMAGE BY HORIZONTAL SCANNING AFFECTS ONLY THE VERTICAL RESOLUTION OF THE IMAGE. THE HORIZONTAL RESOLUTION IS DETERMINED BY THE RAPIDITY WITH WHICH THE ELECTRONICS OR MECHANICS OF THE SYSTEM CAN RESPOND TO ALTERNATIONS OF DARK AND LIGHT IMAGE ELEMENTS WITHIN THE SPACE OF EACH SCAN.

Fig. 9-8. TV resolution.

(A)   (B)

Fig. 9-9. Kell effect.

brightness   The brightness of any display device must be bright enough, yet not too bright, for the user to view it with comfort. Brightness is also closely tied to the contrast between the written and background areas. For example, a darker background requires less brightness in the written portions to fall within the comfortable range.

Any decision as to what the extremities of the brightness comfort range are, would, of course, be a judgment on the part of the viewer. Numerous tests have been run (mostly by the military) attempting to define this range. The tests involve so many variables that it is difficult to relate the results of one test to a different set of variables. Some of those variables that immediately come to mind are, size of display, ambient light, contrast ratio, viewing distance, size of displayed data and faceplate glare or reflection.

In its storage display units, Tektronix specifies brightness at the end of 2000 hours of use at "greater than three foot-lamberts." From all the experience we have had with storage oscilloscopes, some of those being used in data-display application, three foot-lamberts looks like the lower limit of the comfort range under normal ambient light conditions (considering the contrast ratio at the same number of hours). Lower ambient light would allow for less brightness in the display. The eye is so constructed that the center of the comfort range is broad, perhaps from 5 to 60 foot-lamberts, depending on the variables mentioned before.

contrast ratio   The "contrast ratio" of the direct-view bistable tube is defined as the ratio of the brightness of the written portion of the target to the average background brightness. This is different from some contrast-measurement systems that specify contrast as the brightness of the written portions minus the background brightness divided by the background brightness. With the storage tube system, a display that has the written portion the same brightness as the background would have a contrast ratio of 1:1. In the other system that same "contrast" (not contrast ratio) is zero.

Examination of the background of the bistable storage tube discloses that there are some particles which are brightly illuminated (written) though they have not been struck by the write gun. At normal viewing distances, the background appears to be dimly but uniformly illuminated. The number of anomalously fully written particles in the background causes the initial contrast ratio to be about 3:1 instead of a ratio greater than 100:1. As the tube ages, the number of anomalous background particles will rapidly reduce. Thus, relatively quickly, the contrast ratio improves from the order of 3:1 to 5:1 or better. This ratio is felt to be satisfactory since in a normally lit environment, the *reflected light* from the diffuse phosphor screen is of the same order as the background and tends to mask the inherent tube background. Moreover, the diffused reflected room light also produces a *color contrast* between background and written information, enhancing readability. Thus, improvement to better than 5 or 6:1 contrast ratio does dot produce a very significant improvement in performance in a normally lit room.

information display rate

Information display rate is defined as the speed with which a computer display will write dots. It is expressed in dots per second.

The information display rate of the display device is important to the user because it tells him the rate he may write data. We have seen that one of the major disadvantages of the teletype machine is its slow typing speed. Ideally, the information display rate should be fast enough that it doesn't limit the speed of the rest of the system and doesn't annoy the viewer by writing slower than he can read. The direct-view bistable-storage-tube display has an information display rate that meets most of these requirements. It is faster than data can be sent over a voice-grade telephone line, but not faster than data can be sent directly from some fast computers. It can easily write information faster than it can be read.

Information display rate is the reciprocal of the sum of "settling time" and "dot-writing time."

$$IDR = \frac{1}{\text{Settling time + DWT}}$$
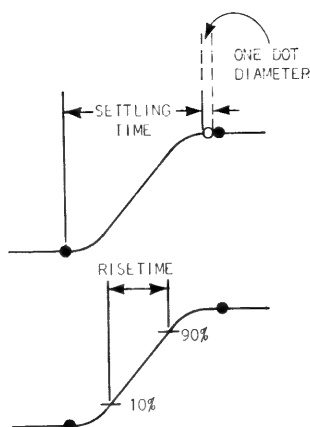
Fig. 9-10. Settling time versus 10-90% risetime.

settling
time

Settling time is the time it takes for the spot to be deflected from its existing position to within one dot diameter from its new location (Fig. 9-10). 10-90% conventional risetime specification is not used because the spot should not be unblanked when the beam is still 20% of the distance from its destination. The spot must be there before the beam is unblanked.

Display unit amplifiers are overpeaked (not gaussian) to provide the fastest settling time. This raises the questions about conventional X-Y and Y-T use which are explained in Chapter 10 on X-Y and Y-T displays.

dot-
writing
time

The writing speed of a conventional cathode-ray tube is usually expressed in centimeters per second of beam-deflection speed, obtained from a visible photograph of a single transient waveform. In the case of the bistable storage cathode-ray tube utilized for computer readout, the data word usually causes the picture to be formed from dots. Therefore, the more useful method of specifying writing "speed" is to specify the *dot-writing time,* which is the length of time the writing beam must be unblanked to store an undeflected spot at the resolution specified. In display units it is a 2000-hour specification. That is, it takes into account the loss of writing speed over 2000 hours of life under "normal usage." "Normal usage" is writing with a random format (not writing in

the same spots again and again) and only 25% stored at
any one time. A typewritten page is typically about
15% written. When interfacing the display to a
computer, it is important that the engineer select a
dot-writing time in the Z-axis drivers that allows for
2000-hour life dot-writing time of the display, so
that the display will be useful for a full 2000 hours.

To understand information display rate more fully,
let's look at the Tektronix Type 611 Storage Display
Unit as an example. The settling time is 3.5 µs per
cm + 5 µs and we want to deflect 22 cm; then the time
would be 77 + 5 = 82 µs. Add to this a dot-writing
time of 20 µs and we need 102 µs to deflect 22 cm and
write a dot. Divide this sum into one and we have an
information display rate of 10,000 dots per second.
In most cases we need not deflect full screen between
each dot. With a shorter deflection distance, settling
time would be less and our information display rate
greater. When writing characters the deflection
distance is very short and the information display rate
approaches 40,000 dots per second.

The 40,000-dots-per-second information display rate is
not slow enough to cause problems when operating the
unit via telephone lines. The ASCII code required 7
bits of data to define a character. At the 2000 baud
telephone transmission rate a character could be
transmitted in *3.5 ms* (7 bits at 0.5 ms per bit). If
the character were being written by a 7 x 9-dot matrix,
the 63 dots would require *1.57 ms* to be written.
[This at an information display rate of 40,000 baud
(25 µs per dot).] If *only* the dots that required
viewing were on for the full dot-writing time, while
the unwritten dots were stepped in 5 µs, as described
in Chapter 8, the character could be written in 0.7 ms.

conventional   Direct-view bistable storage tubes are often used
writing   where the information is not stored as "dots." They
speed   can be used to draw lines, to write characters through
                the use of "strokes" and to do ordinary X-Y lissajous
                work. When used this way the specifications of
                "information display rate" does not convey line-
                writing speed or X and Y bandwidth satisfactorily.
                If "dot-writing time" is broken out of "information
                display rate" we can calculate the line-writing speed.

For instance, if the dot-writing time were 9 μs as it
is on the Tektronix Type 601 Storage Display Unit, the
horizontal resolution 125 line-pairs, and the
horizontal distance 10 cm -- we would have 12.5 line-
pairs per cm; or since there are two dots for each
line-pair, 25 dots per cm. The time to store 1 cm
would be $9 \times 10^{-6} \times 25 = 225 \times 10^{-6}$ or 225 μs per cm.
The writing speed as CRT people specify would be

$$\frac{1}{225 \text{ μs per cm}} = 4.45 \text{ cm per ms.}$$

The actual line-writing speed would be faster than
this due to the precharging of the phosphor by the
moving beam. The charge distribution of the writing
beam on the phosphor is gaussian. There are areas
close to the edge of the stored-spot area which have
some charge from the beam but which are not quite
charged enough to store. If the beam is moving, the
target particles ahead of the spot tend to be
precharged enabling them to store faster than if they
were hit with a beam that was positioned and then
turned on (Fig. 9-11). The Type 601 is then specified
at 5 cm per ms line-writing speed.

If the display being written did not require maximum
resolution, the beam current could be increased. This
would cause a larger line width -- bigger spot size --
and would speed up the writing speed from 2 to 5 times.
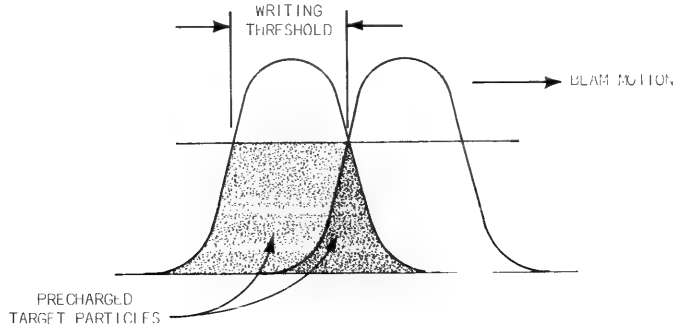
Fig. 9-11. Line-writing speed increased due to
precharging of the target particles.

The Type 611 line-writing speed can be calculated using the same method. The horizontal axis is 16-cm long. The horizontal resolution is 300 line-pairs, so there are 37.5 dots per centimeter (600 ÷ 16). At 20 µs per dot it would take 750 µs to write a centimeter (37.5 x 20). The writing speed as CRT people specify would be $\frac{1}{750 \text{ µs per cm}} \cong 1.3$ cm per ms. Again, these are 2000-hour at rated resolution specifications; a 2 to 5-times improvement can be obtained at poorer resolution.

CRT
life

Of considerable importance to the display terminal user are reliability and maintenance costs of the terminal. The largest single item in this cost is the display CRT. Its life then is important.

Direct-view bistable storage tubes are known to age. The most noticeable effect of aging in the tube is the loss of brightness and writing rate. Less noticeable is the increase in target operating-voltage level. The contrast ratio improves with age; though this is not too noticeable, because room ambient light masks the background.

CRT aging is not fully understood, but it is known to be related to the amount of charge supplied to the stored target. The flood electrons which provide storage are the main source of aging current. Written target areas receive more flood electrons than nonwritten areas and so are aging more rapidly. In the ready-to-write and nonstore-mode aging is almost negligible since flood electrons reaching the screen are greatly reduced.

The storage target is also subject to residual images due to stored images remaining on the screen for periods of time in excess of 15 minutes. A residual image can be either a bright or dark image that is not fully erased by the erase cycle. The image is thought to be the result of trapped stored charges in the target phosphor. Residual images are not too noticeable under high ambient conditions, however, they can be annoying if the room light is subdued or if the CRT has a hood. They usually disappear after a number of erasures.

If there is a repeating pattern to the display, such
as an X-Y axis for graphics, the system should be
programmed to shift axis position each time the image
is erased and rewritten. Another method is to have
the operator move the position control each time the
instrument operating level is readjusted. This will
prevent "differential aging" a phenomenon whereby a
heavily used portion of the target is aged more
rapidly than a nonused portion.

hold
mode

Some direct-view bistable-storage-tube display units
take advantage of a stand-by or "HOLD" mode. In this
mode the flood-gun current to the target is reduced
by duty-cycling the flood guns. Current is reduced
to a level which maintains the stored image at
reduced brightness and prevents residual gas-ion
writing. The operator can, at any time, command
normal flood-gun current and see the image. Thus,
by having the CRT operated at a reduced light output
when it is not being actively observed, one can
significantly extend the life.

In conclusion, direct-view bistable storage tubes
are generally very rugged and are no more subject to
burning than a standard CRT. They *are* subject to
gradual changes in performance. Performance change
can be minimized by: 1) Going to HOLD, NON-store, or
OFF when not in use. 2) Erasing stored information
as soon as it is no longer needed. 3) Running the
writing-gun beam current as low as possible to get
fine traces (also improves resolution). 4) Using the
target uniformly. Try to store information over the
full target area. Don't use a fixed format which
writes over and over at the same spots. 5) If you
must use a fixed format, move registration periodically.

The specifications on Tektronix display unit storage
tubes are for 2000-hour performance. Brightness and
information-display rate will be better than specified
when new.

The most pleasing aspect of the storage cathode-ray
tube is its absence of flicker. Unlike refreshed
displays, the storage tube remains flicker-free even
at very high information density. There is, of course,
no need to reduce the refreshed rate at high
information densities since the storage tube needs no
refreshing. The preferred video is negative, that is,
bright characters on a dark background, since there is

significant background illumination. This mode of
operation allows a relatively uncontrolled background
illumination, i.e. the unwritten portions of the target
may vary considerably in brightness without degrading
the display. In most displays, the total recorded
area occupies much less space than the background,
which is the arrangement for optimal life of the CRT.
When a large area is to be presented as a solidly lit
area, it would be preferred, if at all possible, to
present the large area as a cross-hatched rather than
a solid area.

The relatively low brightness level of this display
requires that the operator sit reasonably close to the
cathode-ray tube in order to benefit from the
available resolution. The resolution capability of
the eye drops off directly with contrast; thus, with
limited tube brightness of the order of 5 foot-
lamberts, in a normally-lit room the contrast ratio is
on the order of 4:1. This limited contrast, although
not fatiguing because the color-contrast is great,
demands that the observer be close to the tube to
discern the very finest detail that the CRT is capable
of displaying.

10

DISPLAY-UNIT CIRCUIT
DESIGN CONSIDERATIONS

The characteristics of, and specifications for, a
display unit depend highly on the cathode-ray tube.
It is the heart of the system. Its design plays a
primary role in determining the design of the rest
of the unit. We have discussed CRT characteristics
and display-unit specifications in Chapter 8. But
the CRT is not, in itself, a display unit. There
must be at least a vertical and horizontal (X and Y)
amplifier to deflect the CRT electron beam, and a
control for turning that beam on and off (Z axis).

For a computer display, the need for large screen,
high resolution, very little drift (positional
stability or repeatibility), low distortion and fast
settling time limits the design of the electronics
of the display. Let's look at the circuit design in
view of these requirements.

deflection   The vertical and horizontal amplifiers of the storage
amplifiers   display unit must have good long-term and short-term
              stability. Since information can be stored for 15
minutes, and new information added any time, drift
(both long or short term) would cause the new data
to be out of registration with the old display. The
refreshed display has a need for short-term stability,
but is not as critical on long-term stability since
the entire display is being renewed each write cycle.

In Tektronix' storage display units the amplifiers
are tightly fedback operational amplifiers. The
feedback minimizes gain drift and assures that a
given voltage will deflect to the same spot repeatedly.
Careful attention to temperature compensation and
power-supply regulation minimizes DC drift.
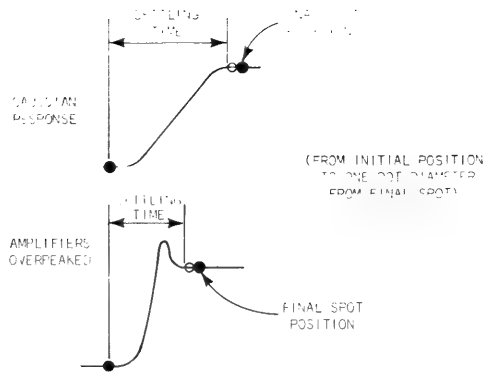
Fig. 10-1. Gaussian versus overpeaked response.

In Tektronix storage display units, amplifiers are
not gaussian -- they are deliberately overpeaked to
provide for minimum settling time. See Fig. 10-1.
Remember from Chapter 9 that the settling time is
the time from initial spot position to within one
dot diameter of the new position. 10 to 90 percent
risetime is not used, because interdot blanking
adjusted for that time would fall 20 percent short
of the necessary blanking time. The amplifier
frequency response is not gaussian, because computer
information is seldom in a form where it is
necessary to preserve the response of a waveform. It
is important that the amplifiers get the cathode-ray-
tube beam from one point to the next as rapidly as
possible. The overpeaked amplifier does this.
Overpeaking does not present any problems in a data-
writing mode, because the beam is blanked during
deflection. The beam is turned on when it reaches
its new dot position. The interest is in *how fast*
the beam gets from point to point -- not *how* it gets
there.

nongaussian
response

The Tektronix Type 601 Storage Display Unit has a
nongaussian response with an amplifier bandwidth of
1 MHz. This is faster than most computers can send
data directly. The Tektronix Type 611 uses magnetic-
deflection amplifiers and is slowed to a bandwidth
of about 45 kHz (1-cm deflection in 8.5 µs -- 3.5 µs
+ 5 µs) for small deflections. This is fast enough
to receive from small computers similar to the Digital
Equipment Corporation's PDP-8I, and much faster than
data can be sent over voice-grade phone lines.

Again, remember, the amplifier bandwidth is not the only consideration for display speed; the writing speed of the storage tube is often the determining factor.

relative
phase
shift

For the user who is not writing data but wants the display for sinewave lissajous and Y-T work, relative phase shift between amplifiers must be considered. The amplifiers are not gaussian, and there is no attempt to keep the relative phase shift the same between X and Y amplifiers, so that there is considerable phase shift before we reach the unit's maximum bandwidth. Most display units have a specification on phase shift between X and Y amplifiers and usefulness for Y-T displays. This specification usually limits the bandwidth to much less than 1 MHz, say 100 kHz.

magnetic
deflection

Magnetic deflection is chosen for some computer display units, for two reasons. (1) Smaller spot size, resulting in better resolution and (2) circuit design advantages for a large-screen display. High-resolution electrostatically deflected systems typically have deflection sensitivities of 50 volts or greater per degree of deflection angle. Assuming at least a 20° deflection angle to keep the tube length reasonable, an electrostatic-deflection amplifier would require deflection voltages in excess of 1000 volts for eight inches of on-screen deflection. Solid-state devices of sufficient bandwidth, that will produce these deflection potentials are not readily available.

The bandwidth of the magnetic system involves two major design considerations, speed versus price and speed versus power. Speed versus price is the usual consideration -- the higher the bandwidth, the greater the price. In addition, the magnetic deflection yoke has some inductance (L) which must be driven to deflect the beam. If many turns are used in the deflection yoke, relatively little current is necessary to deflect the beam. However, the L will be high, causing deflection to be slow, as the speed is inversely proportional to the value of inductance. The opposite is true if few turns are used in the yoke. The speed increases but the driving-current required (and hence the power required) goes up.
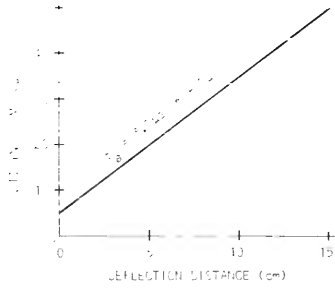
Fig. 10-2. Beam settling time for the Tektronix
Type 611 Storage Display Unit.
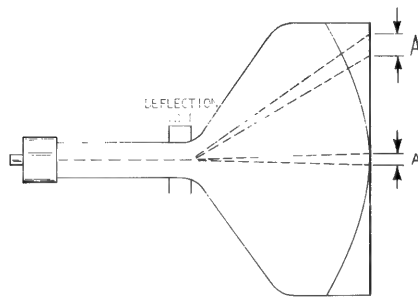
Fig. 10-3. Large-screen, flat-faced, magnetic-linearity
deflection CRT pin-cushion correction needed.
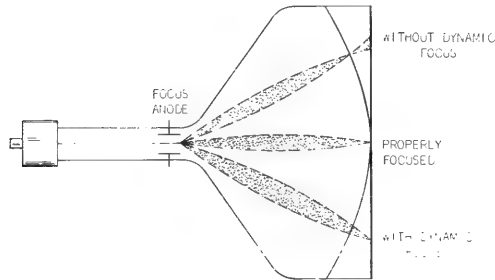
Fig. 10-4. Large-screen flat-faced magnetic-deflection-CRT
dynamic-focus requirement.
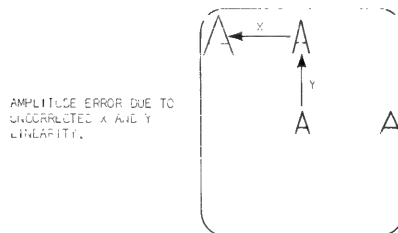
Fig. 10-5. Large-screen-flat-faced-CRT front-view dynamic gain
linearity and pin-cushion correction.

A compromise design results in a deflection system slower than an electrostatic system, but fast enough to accept data at phone-line rates. The current limitation of the yoke drive, dictated by the power versus price consideration, establishes the distance the beam will move in a given time. Thus the "settling time" is a function of distance. Small deflection distances, the most common in display systems, require only small settling times. Fig. 10-2 shows how the settling time varies with deflection distance in the Tektronix Type 611 Storage Display Unit.

Since distance for a given time is "rate," the display is said to be "rate limited." This explains the specification on settling time that states "3.5 µs per cm plus 5 µs." The 3.5 µs per cm is the rate limitation of the amplifier. The 5 µs that is present for long and short deflections is attributed to the resonance of the deflection yoke. The yoke, mostly inductive, has some stray capacitance and is paralleled by a resistor. After being excited, the circuit takes 5 µs for the driving current to transfer from the capacitance to the inductance.

The bandwidth of a "rate-limited" amplifier would depend on "how far" the beam had to deflect; in other words, "how large" the displayed waveform was. For example, the magnetically deflected Tektronix Type 611 Storage Display Unit would have approximately a 6-kHz bandwidth for a full-screen display (20 cm). If only a 4-cm display were used, the X and Y amplifier bandwidth would be about 25 kHz.

large-screen consider-ations

The large-screen, flat-faced, magnetically deflected CRT must have circuits to correct for linearity, pin-cushion distortion and edge defocus. See Fig. 10-3, 10-4 and 10-5.

A character displayed at the center of the screen would be smaller than the same character displayed at the top of the screen, due to the longer beam throw at the top. A character displayed in the corner would suffer from this effect from both vertical and horizontal amplifiers. This change in deflection sensitivity due to position on screen is undesirable. Characters

pin-cushion effect

written at the CRT's extremes should not be larger than those towards the center (pin-cushion distortion). Graphical displays also would be distorted. See Fig. 10-5.

geometry
correction

To eliminate this effect, compression is built into the vertical and horizontal amplifiers; a geometry-correction circuit provides additional correction. The geometry-correction circuit senses on-screen X-Y deflection position and feeds correction signals to both amplifiers; i.e. the correction circuit senses horizontal position and corrects both vertical and horizontal deflection for that position. Likewise, the vertical position is sensed and correction sent to both vertical and horizontal amplifiers.

dynamic
focus

Fig. 10-4 shows that the focus varies depending on position on-screen. To have high resolution, the spot size should be uniformly maintained over the entire screen. A "dynamic-focus" circuit is usually employed in large-screen displays to sense the X and Y position and vary the CRT focus voltage accordingly. In the Tektronix Type 611 Storage Display Unit, the "geometry correction" and "dynamic focus" are incorporated in one unique circuit.

Z-axis
control

After the CRT beam has been deflected to its position, or while it is slowly being deflected, the beam must be turned on or off to make it store or not store. CRT beam on-off is called "Z-axis" control.

analog
Z axis

Conventional oscilloscopes use an "analog" Z axis. The brightness of the display depends on how hard you turn on, or "Z-axis modulate," the CRT beam. The amount of turn-on depends on the amplitude of the Z-axis turn-on voltage. The display is said to have various levels of brightness, called "half-tones."

bistable
Z axis

Since the direct-view bistable storage tube in the storage mode has only two levels of intensity -- stored and not stored -- only two levels of Z axis are required; Z axis "on" and Z axis "off." This simplifies control from the computer, as one bit can be used to write or not-write the beam. For instance, if the bit is a "one" the beam will turn on. If the bit is a "zero," the beam will be off. To make the Z axis compatible with the X and Y amplifiers (+1 V for full-screen deflection), the Tektronix Type 601 and 611 Storage Display Units were set up to respond to +1 volt of signal for turn-on and less than 0.5 V for turn-off. A single transistor stage is usually employed between the digital data and the display Z-axis circuit to convert to the proper voltage levels.

Two circuits are presently used for bistable Z axis in
Tektronix display units. One circuit uses a schmitt
multivibrator, the other a saturated linear amplifier.
The schmitt multi has a slight advantage in "noise
immunity" but is a shade slower in response. The Type
601 uses the schmitt, the Type 611 the linear amplifier.

In the Type 601, the beam is in the "off" position
if the input signal is below +0.5 volt. When the
signal goes above one volt, the multivibrator flips
to its other stable state and the beam turns on.
The beam will not turn off until the Z-axis input
voltage goes below +0.5 volts. The difference between
the turn-on and turn-off levels provides "noise
immunity." For example, suppose the Z-axis input
signal levels used are 0-V "OFF" and +1.2-V "ON."
Should the beam be off, it would take +1 volt of
noise to turn it on. When the beam is on, at least
-0.7 V of signal noise would be needed to turn it off.
Noise immunity is important because display information
could be lost or added if the Z-axis was being turned
on and off by noise.

The Type 611 uses a linear amplifier rather than a
schmitt multivibrator. The on-off specification is
the same as the Type 601, that is, +1 volt or greater
will saturate the amplifier and turn the beam on.
If the input falls below +0.5 V, the beam will be so
low on the amplifier curve that it will essentially
be off. The linear amplifier does not have the
"switching" action of the multi, so there are some
voltages between +0.5 V and +1 V where the beam is
partially on. This means, for example, that when a
0-V input has turned the beam "off" a +0.6-V noise
pulse could turn it on dimly. When the beam is on,
a negative noise pulse could dim the beam to where
it might not store. The differences in noise immunity
and response time of the schmitt and linear amplifier
are almost negligible.

The full-screen deflection of most display units is
in the range of one volt, and the resolution can be
as much as 400 line-pairs. This means that each
addressible line or dot (800) is specified from its
neighbor by an analog voltage of as little as 1.25 mV.
The amplifiers of the display units are designed to
have very little drift, to prevent misplacement of
data.
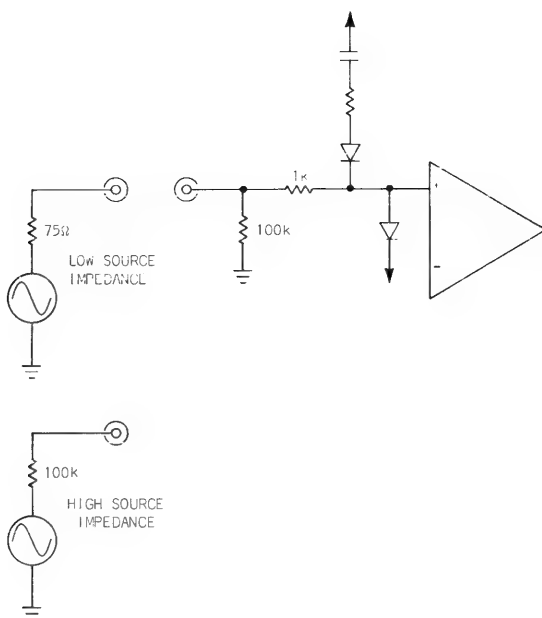
Fig. 10-6. Tektronix 611 Display Unit input
amplifier -- source-impedance
considerations.

source
impedance

The input transistors of the display unit will have
some beta and leakage current change with temperature.
These changes will cause a shift in base current. On
the Tektronix Type 611 Storage Display Unit the zero-
input base current is around 200 nA and changes at
about 5 nA per degree centigrade. If the impedance
of the driving source is large, base-current shifts
will cause positional shifts on screen. For example,
Fig. 10-6 shows the input circuit with two possible
source impedances. (The 1-kΩ resistor in series
with the input is for input protection.) With the
75-Ω source impedance, the base-to-ground impedance
is 1075 Ω (1 kΩ + 75 Ω neglecting the 100-kΩ base
resistor). A 50-nA base-current shift due to
temperature change of 10°C will shift the Type 611
input about 54 μV through the 1075 Ω. This would be
much less than one dot position on screen. With a
100 kΩ source impedance, the base-to-ground impedance
is 50 kΩ. A 50-nA base current would result in a
2.5-mV shift that would move the spot two dot diameters
on screen. This example illustrates why source
impedance should be kept low.

If the source impedance *changes*, then input-transistor
base current through that impedance will cause DC
shifts on screen. For instance, if the source
impedance is 1075 Ω, the 200 nA of base current will
put the zero-input level at 215.0 μV. Should the
source impedance go up to 50 kΩ, the DC level would
jump to 10 mV; a shift of eight dot diameters.

For these reasons, display units specify repeatibility
(positional stability) with a fixed low source
impedance (50 to 1000 Ω).

INDEX

NOTES

This book is but one of a series.
The series consists of two groups,
circuits and measurements. These
texts present a conceptual approach
to circuits and measurements which
apply to Tektronix products.
Several "concept books" are in
preparation; those now available are:

Power Supply Circuits
062-0888-01

Oscilloscope Cathode-Ray Tubes
062-0852-01

Storage Cathode-Ray Tubes and Circuits
062-0861-01

Television Waveform Processing Circuits
062-0955-00

Information Display Concepts
062-1005-00

Semiconductor Devices
062-1009-00

Digital Concepts
062-1030-00